40287

National Library of Canada / Bibliothèque nationale du Canada

CANADIAN THESES ON MICROFICHE / THÈSES CANADIENNES SUR MICROFICHE

NAME OF AUTHOR/*NOM DE L'AUTEUR* VIJAY V. RAGHAVAN

TITLE OF THESIS/*TITRE DE LA THÈSE* EVALUATION OF CLASSIFICATION STRATEGIES FOR DOCUMENT RETRIEVAL

UNIVERSITY/*UNIVERSITÉ* University of Alberta

DEGREE FOR WHICH THESIS WAS PRESENTED/ *GRADE POUR LEQUEL CETTE THÈSE FUT PRÉSENTÉE* Ph. D.

YEAR THIS DEGREE CONFERRED/*ANNÉE D'OBTENTION DE CE GRADE* 1978

NAME OF SUPERVISOR/*NOM DU DIRECTEUR DE THÈSE* C. T. Yu

Permission is hereby granted to the NATIONAL LIBRARY OF CANADA to microfilm this thesis and to lend or sell copies of the film.

The author reserves other publication rights, and neither the thesis nor extensive extracts from it may be printed or otherwise reproduced without the author's written permission.

L'autorisation est, par la présente, accordée à la BIBLIOTHÈQUE NATIONALE DU CANADA de microfilmer cette thèse et de prêter ou de vendre des exemplaires du film.

L'auteur se réserve les autres droits de publication; ni la thèse ni de longs extraits de celle-ci ne doivent être imprimés ou autrement reproduits sans l'autorisation écrite de l'auteur.

DATED/*DATÉ* 1 Sept. 1978 SIGNED/*SIGNÉ* V. Vijay Raghavan

PERMANENT ADDRESS/*RÉSIDENCE FIXÉ* Dept. of Computer Science
Univ. of Regina
Regina, Sask. S4S0A2

NL-91 (3-74)

National Library of Canada

Cataloguing Branch
Canadian Theses Division

Ottawa, Canada
K1A 0N4

Bibliothèque nationale du Canada

Direction du catalogage
Division des thèses canadiennes

NOTICE

The quality of this microfiche is heavily dependent upon the quality of the original thesis submitted for microfilming. Every effort has been made to ensure the highest quality of reproduction possible.

If pages are missing, contact the university which granted the degree.

Some pages may have indistinct print especially if the original pages were typed with a poor typewriter ribbon or if the university sent us a poor photocopy.

Previously copyrighted materials (journal articles, published tests, etc.) are not filmed.

Reproduction in full or in part of this film is governed by the Canadian Copyright Act, R.S.C. 1970, c. C-30. Please read the authorization forms which accompany this thesis.

THIS DISSERTATION
HAS BEEN MICROFILMED
EXACTLY AS RECEIVED

AVIS

La qualité de cette microfiche dépend grandement de la qualité de la thèse soumise au microfilmage. Nous avons tout fait pour assurer une qualité supérieure de reproduction.

S'il manque des pages, veuillez communiquer avec l'université qui a conféré le grade.

La qualité d'impression de certaines pages peut laisser à désirer, surtout si les pages originales ont été dactylographiées à l'aide d'un ruban usé ou si l'université nous a fait parvenir une photocopie de mauvaise qualité.

Les documents qui font déjà l'objet d'un droit d'auteur (articles de revue, examens publiés, etc.) ne sont pas microfilmés.

La reproduction, même partielle, de ce microfilm est soumise à la Loi canadienne sur le droit d'auteur, SRC 1970, c. C-30. Veuillez prendre connaissance des formules d'autorisation qui accompagnent cette thèse.

LA THÈSE A ÉTÉ
MICROFILMÉE TELLE QUE
NOUS L'AVONS REÇUE

THE UNIVERSITY OF ALBERTA

EVALUATION OF CLASSIFICATION STRATEGIES FOR DOCUMENT RETRIEVAL

by

© VIJAY V. RAGHAVAN

A THESIS

SUBMITTED TO THE FACULTY OF GRADUATE STUDIES AND RESEARCH

IN PARTIAL FULFILMENT OF THE REQUIREMENTS FOR THE DEGREE

OF DOCTOR OF PHILOSOPHY

DEPARTMENT OF COMPUTING SCIENCE

EDMONTON, ALBERTA

FALL, 1978

THE UNIVERSITY OF ALBERTA

FACULTY OF GRADUATE STUDIES AND RESEARCH

The undersigned certify that they have read, and recommend to the Faculty of Graduate Studies and Research, for acceptance, a thesis entitled "Evaluation of Classification Strategies for Document Retrieval" submitted by VIJAY V. RAGHAVAN in partial fulfilment of the requirements for the degree of Doctor of Philosophy in Computing Science.

.....
Supervisor

.....

.....

.....

.....
External Examiner

Date ..August 28..1978

To my parents

ABSTRACT

In the context of an automatic document retrieval system it may be necessary to classify documents as well as terms (keywords). Documents are classified in order that the search, in response to a query, may be restricted to only a few most promising subsets of the collection. Term classifications are useful for enhancing the representations of documents and queries which in turn, it is hoped, would increase the chances of the system retrieving all and only the documents that a user is interested in.

It is suggested that the appropriateness of classification strategies to a document retrieval environment may be judged on the basis of factors such as the extent to which it meets the classificatory criterion, the computational efficiency, whether or not the generated clusters depend on the order in which the objects are processed and the degree to which the clusters obtained are sensitive to small errors in the input data. In the light of automatic classification techniques proposed in the literature for the classification of terms and their retrieval effectiveness, it is considered more promising to determine the relationships between terms on the basis of feedback information obtained from the users. Hence, an automatic method that determines the relationships between terms using information provided by the users concerning previous searches is developed. The method identifies

similar terms, in the sense that they are synonymous or substitutable for each other, as well as dissimilar terms that provide substantially different contexts to documents or queries containing them. Strategies that allow for variation in the scope of the term relationships determined are evaluated. Regardless of whether the term relationships are localized or they are global, the retrieval performance when the term relationships are used is better than when they are not. In addition, term relationships that have a narrower scope are found to be more effective in retrieval than their more global counterparts. The relationships between terms that occur in a small number of documents are found to be more significant for retrieval purposes than those of terms occurring in a large number of documents.

The term relationships are incorporated into the retrieval process by using a generalized similarity function to measure the closeness between a query and a document. The function consists of a term match component reflecting the extent to which the index terms match, a positive component which measures the similarity between the terms in the document to the terms in the query, and a negative component which provides a measure of the dissimilarity between the terms in the two items. A multivariate statistical method known as discriminant analysis is shown to be useful in determining the relative importance of the components used in the similarity function.

Stability (that is, a classification generated being not too sensitive to small errors in input data) is one of the desired properties of a clustering method. While many of the currently available classification techniques have been evaluated on factors such as retrieval effectiveness and computational efficiency, very little research has gone into the assessment of stability. Consequently, this aspect of a number of commonly used graph theoretic clustering schemes is analysed.

It is shown that, in terms of the measure of stability used, any "reasonable" cluster definition which has the property that a cluster will be generated for any two sufficiently close objects cannot be more stable than the "connected component" definition. A number of classification schemes are found to have the above property. Clusters defined as connected components are shown to be the most stable, while those defined as "maximal complete subgraphs" are found the least stable among the cluster defining methods characterised by the property mentioned above. The stability of clusters defined as connected components is bounded.

ACKNOWLEDGEMENTS

I will always owe an intellectual debt to Clement Yu who has been so instrumental in bringing this work to fruition. I have benefited as immensely from his insights and intuitions on problems as I have from his criticisms, particularly of my writing. It is a pleasure to work with Clement as he has been able to provide an excellent research environment and has always kept his doors open for us (literally too!).

I am very grateful to Drs. Cabay, Chen, Davis, Korfhage and Sampson for taking the time to read, criticize and comment on this thesis.

Many of the experiments carried out in this thesis would have been impossible without the help of Ron Senda, the IBM-360 Assembler wizard (among other things). Mr. Ewasechko and Mr. Prat of the Operations Group obliged me by providing an excellent turn-around on many system-boggling computer runs. Ella Ritz and Marilyn Butchko of Data Entry Services did a superb job of typing parts of this thesis. Paul Kam helped in the preparation of the manuscript for a paper that is based on some of the results contained in this thesis. I am very thankful to all these people.

I gratefully acknowledge financial assistance received from the Department of Computing Science and a grant from National Research Council of Canada to Dr. Yu.

Last, but not the least, I wish to thank Bernadette, Nalini and Sanjay, the other Raghavans, for providing moral support and encouragement all through this endeavour and for so graciously allowing me to spend so much time, that is rightfully theirs, away from them.

TABLE OF CONTENTS

Chapter	Page
1 INTRODUCTION	1
1.1 Classification	1
1.2 Document Retrieval	5
1.3 Strategy for Evaluating Classification Techniques for Document Retrieval	7
1.4 Focus	9
1.5 Preview	13
2 TERM CLASSIFICATION FOR DOCUMENT RETRIEVAL SYSTEMS	16
2.1 The Environment	16
2.2 The Problem	17
2.3 Approaches to Term Classification	18
2.3.1 Measurement of Similarity	19
2.3.2 Graph Theoretic Concepts Related to Clustering	20
2.3.3 Automatic and Semi-automatic Techniques for the Classification of Terms	23
2.3.3.1 Methods Based on Coocurrence Hypothesis	23
2.3.3.2 Other Methods	27
2.4 Motivation Behind the Method to be Proposed for Determining Term Relationships	32
3 A NEW METHOD FOR DETERMINING THE RELATIONSHIPS BETWEEN TERMS	34
3.1 Description of the Method	34
3.1.1 The Hypothesis	34
3.1.2 The Proposed Method	38

Page

3.2 The Experimental Design 41
3.2.1 Test Collections 41
3.2.2 Experimental Specifications 41
3.3. Basic Experimental Results 45
3.3.1 The Effect of Weight Functions 45
3.3.2 The Effect of Document Frequencies of Terms 50
3.4 Experiments With Clustered Queries 53
3.5 The Determination of the Coefficients in the Similarity Function 56
3.5.1 Motivation 56
3.5.2 Problem Specification 58
3.5.3 Approach 58
3.5.4 Experimental Results 61
3.6 Statistical Significance of the Experimental Results 67
3.7 Computing Time 69
4 METHODS FOR EVALUATING THE STABILITY OF CLASSIFICATION STRATEGIES 71
4.1 The Need for Stable Classifications 71
4.2 Difficulties in the Evaluation of Stability 73
4.3 Comparison of Classifications 74
4.4 Motivation for the Approach Selected for Stability Analysis 81

Page

5 STABILITY ANALYSIS OF CERTAIN GRAPH THEORETIC CLUSTERING METHODS 82

5.1 Introductory Concepts Relating to the Measure of Stability 82

5.1.1 Preliminary Definitions on Graphs 82

5.1.2 Notations Relating to the Measure of Stability 84

5.2 Stability Analysis of Certain Cluster Defining Methods 85

5.2.1 Cluster Characteristics 85

5.2.2 The Main Result Concerning "Adjaceny Oriented" Clustering Methods 91

5.2.3 Stability Ordering in Adjacency Oriented Families of Clustering Methods 96

5.3 The Maximum and the Minimum Amount of Work for Clusters Defined as the CC's 100

5.3.1 The Lower Bound 101

5.3.2 The Upper Bound 105

5.5.3 A Comparison of the Bounds 113

6 SUMMARY OF FINDINGS AND PROPOSALS FOR FURTHER RESEARCH 116

REFERENCES 122

Page

APPENDIX-A EXPERIMENTAL DETAILS FOR THE PROPOSED METHOD 129

A.1 Determination of the POS and NEG Counts 129

A.2 The Strategy for Selecting the Evaluation Set of Queries 129

A.3 The Incorporation of the Relationships Between Terms into the Retrieval Process 130

A.4 The Weighting Functions W_1 and W_2 131

A.5 The Weighting Function W_3 135

A.6 Method of Comparing Different Retrieval Strategies and the Computation of Statistical Significance 136

APPENDIX-B TECHNICAL RESULTS RELATING TO STABILITY ANALYSIS 139

LIST OF TABLES

Table	Description	Page
1	Average precision for W_1 and W_2 (ADINUL)	46
2	Average precision for W_1 and W_2 (CRN4NUL)	47
3	Average precision for W_3 (ADINUL)	51
4	Average precision for W_3 (CRN4NUL)	52
5	Average precision when query clustering stragegy based on a simple match function at two cutoff levels is used with W_3 (CRN4NUL)	55
6	Average precision when query clustering strategy based on a nearest neighbour criterion is used with W_3 (CRN4NUL)	57
7	Standardized coefficients for two runs using all three components in discriminant analysis (the sizes of samples from the two groups are about equal)	64
8	Standarized coefficients for two runs using only the positive and negative components in discriminant analysis (the sizes of samples from the two groups are about equal)	65
9	Average precision under mode 1 for W_1 when i) all a_i's and ii) only a_2, a_3 are obtained using discriminant ANALYSIS (CRN4NUL)	66

LIST OF FIGURES

Figure		Page
1	An illustration of some cluster types	24
2	The distributions of A'X for DR pairs in which D is relevant to R, and where D is not relevant to R	60
3	A dendrogram to illustrate the computation of cophenetic values	76
4	An initial and a perturbed graph to illustrate D_t is not stability ordered	98
5	Stability characteristics of graphs in $\mathcal{G}(s,e,n_1,n_2,\ldots n_s)$ for d deletions, a' inter-cluster additions and (a-a') intra-cluster additions	115
6	Representation of term pairs that are significantly positively or negatively related (Unshaded region)	132
7	Illustration of the variables in the weighting functions	133

Chapter 1

INTRODUCTION

Plato had defined Man as an animal, biped and featherless, and was applauded. Diogenes plucked a fowl and brought it to the lecture-room with the words "Here is Plato's man". In consequence of which there was added to the definition, "Having broad nails".

DIOGENES LAERTIUS, Lives of Eminent Philosophers.

1.1 Classification

Classification encompasses many diverse techniques for discovering structure within complex bodies of data. In a typical example one has a sample of data units (persons, organisms, documents) each of which is represented by a finite number of attributes or features. The objective is to place the data units or the attributes into groups such that the objects (or entities) within a cluster are more strongly related to each other than those in different clusters.

In this sense, the term classification refers to a process. The terms clustering and cluster analysis have also been used to refer to the same process. The need for cluster analysis arises in a natural way in many fields of study such as life sciences, social sciences and information sciences. The term classification has been employed, in some disciplines, in the restricted sense of putting entities into distinct classes as opposed to arranging them in a

continuous spectrum, cline or some other ordination showing no distinct divisions. In this thesis the term is not restricted in this manner. If a group of objects are identified as being related then such a group is referred to as a class or a cluster. The set of classes or clusters constitutes a classification. Thus the result of classification is a classification. In the placement of objects into clusters, it is generally required that the classification generated be exhaustive of the objects under consideration and that no class in the clustering includes any other class. If the overlap between any two classes in such a classification is null, then it is called a partition.

Problems to which methods of classification have been applied fall roughly into two categories depending on whether the primary objective of classification is to provide a simplification and summarization of the data or it is to place objects in classes so as to optimize some classificatory criterion developed from a knowledge of the application environment (Jardine, 1970). Where the objective of classification is to obtain a summarization, in which the individual differences between the objects to be classified are suppressed in order to derive more general properties which are characteristic of groups of objects, usually some "intrinsic" classificatory criterion is employed (Jardine and Sibson, 1971). This is because, for such problems, it difficult to choose simple "extrinsic" criteria by which the

results of the classification can be evaluated. Furthermore, extrinsic criteria in these cases may themselves be largely matters of opinion. Some intrinsic criteria have been proposed (see Sneath and Sokal, 1973). In essence, these criteria are measures of how well the classes obtained reflect the relationships between objects (that is, the information from which the classification is derived).

In the latter case, there exists an extrinsic criterion of the 'goodness' of a solution. The nature of the extrinsic criteria is different depending on the type of application at hand. In certain situations, there may exist an a priori classification which a classification method is intended to reconstruct in detail. The Platonic concept of error in the assignment of objects to classes is applicable in such cases. Thus, it may be possible to achieve a probablistic assignment of objects so that the properties of particular objects are inferrable from a knowledge of the classes to which each object belongs. In other cases, certain criterion functions may be used to precisely specify the particular manner in which the entities within and between clusters must be associated in the resulting classification. The complexity of the classification procedure will, of course, depend on the characteristics of the function and the existence and the uniqueness of an optimum.

A more common approach to judging the validity of a method of classification is simply to ask 'How well does it

work?'. In the study of biological organisms, for instance, it may be important that a classification derived be useful in testing evolutionary hypothesis derived from other data or from theoretical considerations (Sneath and Sokal, 1973). The objective of conventional library classifications (Dewey, UDC, etc.) has been to place documents relating to the same subject area in one class in order to minimize the effort required for search.

Numerous methods for cluster analysis have been proposed. The earliest attempts at classification were based, as Cain (1958) has shown, on Aristotelian logic. This method was used by early workers such as Cesalpino (1519-1603) and even largely by Linnaeus (1707-1778). Discussions of many recent methods can be found, among other places, in Bonner (1964), Ball (1965), Watanabe (1965), Johnson (1967), Lance and Williams (1967a,b), Cormack (1971), Jardine and Sibson (1971), Anderberg (1973), Hubert (1974), Yu (1974b) and Matula (1977).

The focus of this thesis is on <u>automatic</u> classification techniques that are of interest to document retrieval systems. Classifications obtained by means of automatic methods are based on some measure of similarity between the objects as determined by the attributes they possess. A method, however, may or may not require that the computation of the similarity measure between every pair of objects be the first step in the identification of classes. If this

step is needed, then an object - object similarity matrix whose $(i,j)^{th}$ element represents the degree of closeness between the i^{th} and the j^{th} object is created. Note that cluster methods based on such a similarity matrix require at least $O(n^2)$ operations, for n objects.

1.2 Document Retrieval

The field of information storage and retrieval encompasses a broad scope of topics ranging from basic techniques for accessing data to sophisticated approaches for the analysis and understanding of natural language text. According to Minker (1977), three general areas of investigation can be distinguished within this field.

1) Document Retrieval
2) Generalized Data Management
3) Question - Answering

Document retrieval systems deal with the storage, maintenance, indexing and retrieval of documents. These have been termed reference providing systems, where the system is required to refer a user to a list of documents that are likely to be of interest to him, in contrast to data providing systems concerned with the retrieval of information explicitly stored or facts derivable, on the basis of general rules or axioms, from such information (Salton, 1975b). In other words, data providing systems respond to specific requests for data with specific answers containing, as far as possible, only the data actually

requested. Reference providing systems, on the other hand, also service users that might be interested in a state-of-the-art report for a particular subject area. In this sense, data providing systems include question-answering as well as generalized data management systems.

In a document retrieval system, input data are generally represented in natural language text form. When such systems are mechanized, it is necessary to represent the documents in a manner that facilitates efficient retrieval. Usually, each document is represented by a set of descriptors. A descriptor may be a term, phrase, or a term class. For example, a document on "Approaches to the design of automatic information storage and retrieval systems" may be represented by the following descriptors:

(automatic, approach, information system, processing).

In this representation "automatic" and "approach" are simple terms, "information system" is a phrase and "processing" is a term class that includes "storage" and "retrieval". The process of selecting such content descriptors is referred to as indexing. Indexing may be accomplished by automatic, semi-automatic, or manual means (Salton, 1975b). Documents may be classified so that the search effort can be restricted to the most promising groups of documents.

Depending upon the system design objectives it may be desirable to accept natural language input requests. Most

document systems search for the descriptors contained in the query and do not subject the query statement to syntactic or semantic analysis. More specifically, the system, in response to a question, generally compares the descriptors in the query with those in the document description, and then orders the documents based on some measure of closeness that reflects the degree to which each document pertains (as judged by the system, but not necessarily by the user) to the subject matter of the query. The output of a document retrieval system may be a list of addresses or names of documents, a listing of some surrogate (e.g. abstract) or the entire texts of selected documents.

1.3 Strategy for Evaluating Classification Techniques for Document Retrieval

There are a number of factors that determine the suitability of a classificatory strategy to a given application. The most important of them is, of course, the classificatory criterion. A method that satisfies the clustering criterion is considered effective. The efficiency of the clustering algorithms is also very important. Here one is concerned with the cost of generating a classification.

In addition to these factors, classification strategies may also be subject to various other constraints. Jackson (1969a, 1969b and 1971) has suggested that it is desirable

that clustering algorithms possess the following properties:

a. Order independence: The classification obtained is independent of the order in which the objects are treated.

b. Stability: Small changes in the material to be classified lead only to small changes in the resulting classification.

c. Independence of scale: The classification generated is unaffected by multiplication of the similarity matrix by a positive constant.

When large amounts of data are involved, errors may be made in the choice of the parametric representations for the objects being classified and in the conversion of this data into machine readable form. If the clustering algorithms employed are stable, then an appropriate classification can be obtained in spite of these errors. The scale independence property is desirable since, in many applications, the range of values that the attributes describing the objects can assume and the scale used for the measures of similarity between objects are essentially arbitrary.

An order independent clustering method yields a unique and well defined set of classes. However, in order to ensure that the result of classification is independent of the processing order, it may be necessary to assess all subsets of the objects with respect to the classificatory criterion. Clearly, this is a very time consuming process.

Consequently, in devising classification strategies, order dependence may be tolerated in order that the process be completed in a reasonable amount of computing time.

In this thesis, certain classification methods are evaluated in terms of the factors and constraints specified above.

1.4 Focus

In the context of document retrieval systems it may be necessary to classify documents as well as terms. The process of classifying terms is usually referred to as thesaurus construction (Salton, 1975). The output of a document retrieval system, which is an ordering of the documents in terms of their relevance to a query, is also, according to the definition presented in section 1.1, a type of classification. In this sense, the ultimate objective of a document retrieval system is one of obtaining an ordering in which each document relevant to a query has a higher rank than any document not relevant.

The need for the classification of terms may arise when the documents and the requests are represented by only a set of keywords. It is quite common in such an environment to determine the closeness between a document and a request on the basis of terms that are in common. There may, however, be disagreements between the user and indexer in regards to the terms they choose to denote particular concepts.

Consequently, the system may sometimes retrieve documents that the user is not interested in and, at other times, may not retrieve certain documents that the user would have considered useful. Under these circumstances one would expect the performance of the document retrieval system to be improved if the measure of closeness is based not only on terms in common but also on those that are related or similar. Consequently, a number of automatic methods have been proposed for the construction of term classes (Stiles, 1961; Guiliano and Jones, 1963; Needham and Sparck-Jones, 1964; Abraham, 1965a; Stevens et al., 1965; Doyle, 1966; Salton, 1966; Dattola and Murray, 1967; Lewis et al., 1967; Needham, 1967; Vaswani, 1968; Lesk, 1969; Auguston and Minker, 1970b; Sparck-Jones, 1971, 1973; Minker et al., 1972). These methods are based on the hypothesis that the more often two terms tend to cooccur, the more likely they are to be inter-substitutable for each other. That is, the measure of closeness between two terms is specified as a function of the number of documents in which they cooccur. Clearly, the assumption implies that the terms belonging to a particular group need not be synonyms, or even near-synonyms, and they need not be generically related either. In other words, terms which are related in the sense that they refer to the same topic or collection are, for retrieval purposes, acceptable substitutes for one another.

Jackson (1970, 1971) and Yu (1974a, 1975) proposed classification strategies that are based instead on the

relevance judgments provided by the users. In these and also in the strategies based on the cooccurrence hypothesis only terms that are similar or synonymous to each other are identified. In this thesis an efficient method for determining the relationships between terms, which is based on user relevance judgments, is proposed. The method, in addition to identifying the degree of similarity between terms, also determines the extent to which two terms are dissimilar. If a term pertains to a different topic or usually characterises a context different from that of another term, then the terms are considered dissimilar. The effectiveness of the term relationships obtained is tested in a carefully designed experimental environment.

A great many of the methods for the construction of term classes referred to in this section are graph theoretic. These methods, to begin with, transform the object-object similarity matrix into a binary matrix. That is, a threshold is applied and the value of the $(i,j)^{th}$ element is made 1 if the corresponding similarity value is greater than or equal to the threshold; the element is made zero otherwise. Let this matrix be referred to as the object-object adjacency matrix (or, simply, adjacency matrix). A graph is then associated with the adjacency matrix where each vertex in the graph corresponds to an object, and an edge is associated between two vertices if the element in the adjacency matrix for the corresponding objects is a 1. Clusters are then identified by choosing one

of various graph theoretic schemes available for specifying the conditions under which two vertices should be placed in the same cluster. This approach has the advantage that nearly all the schemes employed generate order independent classification. It should be noted, however, that the computation of the object-object similarities itself requires a processing time of $O(m^2)$, where m is the number of terms in the indexing vocabulary. In contrast, the method proposed in this thesis for obtaining the relationships between terms is order dependent but is more attractive from the point of view of efficiency.

Many researchers have evaluated graph theoretic classification strategies (for document as well as keyword classification) from the point of view of their effectiveness in retrieval (Bonner, 1964; Dattola and Murray, 1967; Gotlieb and Kumar, 1968; Vaswani, 1968; Auguston and Minker, 1970b; Sparck-Jones, 1971; Van Rijsbergen, 1971; Minker et al., 1972). Efficient algorithms have also been developed for identifying the clusters that correspond to the various graph theoretic schemes (Bonner, 1964; Auguston and Minker, 1970a; Cole and Wishart, 1970; Sparck-Jones, 1971; Jardine and Sibson, 1971; Mulligan and Corneil, 1972). However, only a few researchers have considered the evaluation of such methods from the point of view of their stability (Jackson, 1969a, 1969b; Yu, 1976; Corneil and Woodward, 1978).

Jackson's (1969a) work is limited in the sense that he does not measure the change in the classification corresponding to some change in the material classified. More precisely, Jackson assumes that the objects are represented by binary attributes and he proposes a function which estimates the changes in the similarity values corresponding to certain changes in the object-attribute binary matrix. He then concentrates on efficient algorithms for computing the proposed function. In his other study (1969b), Jackson proposes a measure which reflects the extent to which a classification accurately represents the data from which it is derived. Although it appears that this approach can be applied to the problem of measuring stability, it is not clear if such usage would be appropriate. Corneil and Woodward's approach is interesting but their experimental findings are rather limited.

The approach suggested by Yu to measure the disturbance in classification due to small changes in input data, however, is found to be well suited for formal treatment. In this thesis a number of well known graph theoretic clustering schemes are compared analytically using the measure proposed by Yu.

1.5 Preview

In Chapter 2, the motivation for the classification of terms is presented in detail and a few of the well known

techniques are reviewed. In the light of the experiments conducted and results obtained by other workers a promising venue for research is identified. This leads to a new method for determining the relationship between terms.

The details of the proposed method for obtaining the term relationships are presented in Chapter 3. The retrieval performance obtained when the term relationships are incorporated into the retrieval process are compared to that of using only the term match information. It is found that the method proposed is effective.

Chapter 4 considers the need for and the difficulties in the measurement of stability. An account of the research in this area is then presented. The motivation for the approach used in this work for measuring the disturbance in classification due to small changes in the input data is provided.

In Chapter 5, a number of graph theoretic classification strategies are identified as being "adjacency oriented". Clusters defined as connected components[1] and maximal complete subgraphs[1] are both found to be adjacency oriented. Connected component definition is shown to be the most stable and maximal complete subgraph definition the least stable of all adjacency oriented clustering strategies. An upper and lower bound for the stability of

---

[1] Defined in Chapter 2.

clusters defined as connected components, corresponding to a specified amount of perturbation, is also derived.

Chapter 6 consists of a summary of the results obtained and includes proposals for further research.

Chapter 2

TERM CLASSIFICATION FOR DOCUMENT RETRIEVAL SYSTEMS

2.1 The Environment

In general, the descriptors used to represent documents and requests may be terms, phrases and/or classes of related terms. Since we are considering approaches to the construction of term classes, an environment in which the documents and the requests are indexed only by terms or keywords is assumed. Thus, the documents and the requests may be logically viewed as sets. This view is taken in some parts of this thesis. It is, however, more convenient to view documents, requests or terms as vectors in a multidimensional space when similarity functions are being described. (The particular view taken will become clear in context; where this is not the case the view taken will be stated explicitly.) These vectors may be binary or weighted. In the representation of a document, for instance, a particular element may be 1 or 0 indicating the presence or the absence of the corresponding term, or it may be a weight that reflects the importance of the term to the document. The retrieval process, as mentioned earlier, is based on a similarity function that measures the closeness between a document and a query, and a threshold value. A document is termed relevant in relation to a user query if the document is of interest to the user; it is irrelevant, otherwise.

2.2 The Problem

In such a system the measure of similarity is usually a function of the number of terms in common between a query and a document. Since relevance judgments are externally specified, it is often the case that a number of documents relevant to a given query are not retrieved by the system. Similarly, some of the retrieved documents may not be relevant to the query. Such deviations could arise even if the queries and the documents are characterized reasonably well by the keywords. A possible explanation for this behaviour is that those documents relevant to but not retrieved by the query are represented by keywords that are distinct from, but having a meaning similar to those of the query. For instance, consider a query and a document represented by the following terms:

QUERY = (design, automatic, information, retrieval, approach)

DOCUMENT = (computerized, principles, text, retrieval, systems, development)

It seems very likely that the document is relevant to the request, but there is only one term in common between the document and the query. So, in this case the document does not score a high enough correlation for it to be retrieved. We can raise the correlation to the desired level if it is possible to identify terms such as (computerized, automatic) and (design, development) to be semantically

related. Consequently, a lot of research effort has gone into the problem of term classification. The idea is to identify classes of terms within which terms are similar or related to each other. The range of request-document matches obtainable using the terms alone could, then, be extended by suitably incorporating the knowledge gained from the classification into the retrieval process.

2.3 Approaches to Term Classification

The first suggestions for the construction of term classifications were put forward in the late 50's when work in this, and in the complementary area of automatic document classification was begun by Borko, Doyle, Guiliano, Needham and Stiles. A survey of this early work is reported in Stevens et al. (1965).

Of the more recent investigations, the results of Cleverdon et al. (1966) and Salton and Lesk (1968) deserve specific mention. Cleverdon's study of manually constructed thesauri for the Cranfield collection show that the keywords alone worked at least as well as the manual thesauri used in that project. But Salton and Lesk found that if a manual thesaurus for this collection was constructed more carefully better performance could be obtained than for terms alone. Specifically, they formulated certain guidelines or rules to which the term classes created are required to conform. For instance, ambiguous terms such as "bat" were entered into

only those term classes that were likely to be of interest in the subject area under consideration. (That is, the term would not be encoded to represent an animal if the document collection deals with sports and ball games.) For another example of a rule, each term that appeared in a large number of documents was placed into a class by itself rather than being combined into classes with other terms. The term classes obtained in this manner consisted predominantly of synonyms and near-synonyms.

Before the work in automatic techniques for term classification is reviewed, it is convenient to describe various methods that are employed in order to measure the similarity between terms and introduce a few graph theoretic concepts that are needed throughout this thesis.

2.3.1 Measurement of Similarity

By far, the most popular premise on which the closeness between terms is judged is that the terms that tend to cooccur often are likely to be related. It is clear from the assumption that the terms eventually assigned to a particular group may not be synonyms or even near synonyms and they may not be generically related either. However, this premise is quite valid in the light of the classificatory criterion that, for retrieval purposes, terms within a class be substitutable for each other. In other words, terms that pertain to the same topic - such as

"boundary" and "layer" which are commonly used in the subject of aerodynamics - may be acceptable substitutes for one another. Similarity functions commonly used to compute the closeness between terms are considered next.

Let $A = (a_1, a_2, \ldots a_n)$ and $B = (b_1, b_2, \ldots b_n)$ represent two terms (If each row of document-term matrix constitutes the vector representation of a document, then the columns of the matrix correspond to the vector representation of terms). The following is a list of some similarity functions that have been used by different researchers for the purposes of term classification:

a) Cosine function:

$$\text{Cosine}(A,B) = \sum_{j=1}^{n} a_i.b_i \;/\; \text{SQRT}\left[\sum_{j=1}^{n} a_i^2 \cdot \sum_{j=1}^{n} b_i^2\right]$$

b) Logical overlap function:

$$\text{LO}(A,B) = \sum_{j=1}^{n} \min(a_i,b_i) \;/\; \min\left(\sum_{j=1}^{n} a_i, \sum_{j=1}^{n} b_i\right)$$

c) Tanimoto function:

$$T(A,B) = |A \cap B| \;/\; |A \cup B|$$

where A and B denote the set of documents containing, respectively, term A and term B[2].

2.3.2. Graph Theoretic Concepts Related to Clustering

A threshold can be applied to the term-term similarity matrix to obtain an adjacency matrix. Clusters can then be identified by applying one of the many graph theoretic

---

[2] $|X|$ is the cardinality of set X.

concepts of relatedness to the graph associated with the adjacency matrix. A few such concepts of relatedness are introduced next.

Definition 2.3.2.1: Let V be a set of elements. The unordered product of V with itself, denoted by V & V, is the set {v & w | v, w ∈ V and v & w is the same as w & v}; i.e., V & V is the unordered Cartesian product of V with itself.

A graph, G = (V,E), is a set of vertices V, a set of edges E and a mapping Φ :E -> V & V. If e ∈ E and v, w ∈ V such that Φ (e) = v & w, then v and w are said to be adjacent.

In relation to the physical problem, the set of objects to be classified corresponds to the vertices of the graph. If the similarity between two objects is greater than or equal to a threshold, then the associated graph has an edge between these two objects.

A subgraph, $G_1 = (V_1, E_1)$ of a graph G = (V,E), is a graph such that $V_1 \subset V$, and $E_1 \subset E$.

Definition 2.3.2.2: Let P be a property. A maximal set S having property P is a set such that $S \cup \{x\}$ does not satisfy P for any $x \notin S$.

Definition 2.3.2.3: A maximal complete subgraph (MCS), $G_1 = (V_1, E_1)$ of a graph (V,E), is a maximal subgraph of G, such

that for every x_1, $x_2 \in V_1$, x_1 is adjacent to x_2.

In other words, every object contained in a maximal complete subgraph is very close to every other object in that subgraph.

Definition 2.3.2.4: Two vertices v and w are connected if one of the following is satisfied.

(i) v is adjacent to w; or v is w

(ii) there exists a set of vertices x_1, x_2, ..., x_p such that v is adjacent to x_1, x_i is adjacent to x_{i+1} for $1 \leq i \leq p-1$ and x_p is adjacent to w.

When two objects are connected but not adjacent, it is very likely that the objects have some common attributes as indicated by their closeness to other objects. On the other hand, objects that are not even connected to each other should be considered not related in any significant way.

Definition 2.3.2.5: A connected component (or, simply, component and abbreviated as CC) $G_1 = (V_1, E_1)$ of a graph $G = (V,E)$ is a maximal subgraph of G such that for every x_1, $x_2 \in V_1$, x_1 and x_2 are connected.

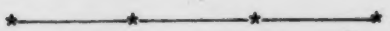
The definition indicates that each connected component represents a largest possible set of related, but not necessarily very close, objects.

2.3.3 Automatic and Semi-automatic Techniques for the Classification of Terms

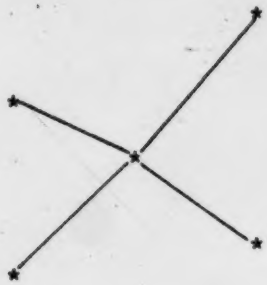
2.3.3.1 Methods Based on Cooccurrence Hypothesis

Sparck-Jones (1971) and Sparck-Jones and Jackson (1967) have carried out a rather extensive set of experiments on automatic term classification. Although, in some of their experiments, a weighted form of Tanimoto function has been used to obtain a term-term similarity matrix, their work is based primarily on the simpler unweighted Tanimoto function described earlier. Clusters are identified as stars, strings, MCS's, or clumps. These have been illustrated in Figure 1. For strings, one starts with an object, finds any object adjacent to it and then a further object adjacent to the second and so on until some terminating condition is met; alternately, rather than any object, the object most similar to the current one may be chosen. Stars are created by identifying a certain specified number of objects each of which is adjacent to an initial object. Clumps, on the other hand, depend on a greater density of edges within the set than outside. More comprehensive discussion of these cluster indentifying schemes can be found in the references cited.

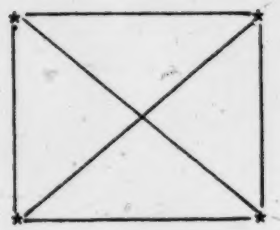
Sparck-Jones and Jackson have also looked at a number of different strategies for incorporating the term classes into the descriptions of documents and requests. They conclude that, for the collection tested (Aslib-Cranfield collection of 200 documents), better performance than for



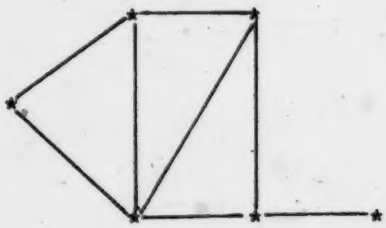
STRING



STAR



MCS



CLUMPS

FIGURE 1. An illustration of some cluster types.

terms alone can be obtained by using automatically constructed term classes. Their work recommends the use of unweighted Tanimoto function with a high threshold; the classification given by strings, small stars, high threshold MCS's and certain clumps; and simple class matching mode of retrieval. They also suggest that only non-frequent terms should be allowed to be classified.

Vaswani's (1968) method consists of modifying the graph obtained from the initial data in a certain way to obtain a new graph and then defining clusters as the set of MCS's of this graph. Among the pairs of objects not adjacent in the initial graph, suppose S_1 denotes the set of pairs of objects occurring inside "potential" clusters and S_2 the set containing the pairs that do not. It is, then, hypothesized that the proportion of pairs in S_1 that are connected by paths of length 2 (i.e. via one other object) is higher than the corresponding proportion for S_2. Consequently, given the initial adjacency matrix, A, a new object-object similarity matrix is obtained by computing $A' = a.A + b.A^2$ (where a and b are scalars and A and A^2 are formed with zero diagonals). Intuitively, a weight having two components is assigned to each pair of objects: one indicating whether or not the objects are adjacent and the other being proportional to the number of paths of length 2 between them. By applying a suitable threshold to A' a new adjacency matrix that has the same number of edges as A (this process may be thought of as providing a redistribution of the edges in the initial

graph) is obtained.

This whole process is repeated for a predetermined number of iterations. The MCS's of the last graph produced are said to form the clusters. The clusters so obtained for a test\collection are found to consist of seemingly (as judged by inspection and intuitive assessment) related terms. The author's (jointly with Cameron, 1970) subsequent retrieval experiments on a collection of 11,500 document abstracts in computer engineering show that simple means of comparing keyword stems provides very good level of performace in relation to the results obtained when the term classes created by his method are used.

Auguston and Minker (1970a, 1970b) and Minker et al. (1972) have also evaluated graph theoretic approaches to term classification. In these studies the term-term similarity matrices are based on Tanimoto function. Auguston and Minker's studies consist of analysing the characteristics of clusters obtained using the CC definition, the MCS definition and a scheme for merging MCS's into broader clusters (like that of Gotlieb and Kumar, to be described in the next section). The effect on the retrieval performance of such clustering methods is investigated in Minker et al. For these experiments only the CC and the MCS cluster definitions are employed. The term class information is incorporated into the retrieval process by the expansion of queries; that is, terms that belong to

the same class as the terms already in the query are appended. A scheme is proposed for assigning weights to the terms added. Retrieval experiments are performed on 6 different document collections (3 types of IRE collection and 3 types of MEDLARS collection).

Minker et al. conclude that, when retrieval is carried out using automatically expanded queries, the only significant changes observed in the overall retrieval performance are degradations, although some small improvements over limited portions of "recall" range can be realized in a few isolated instances. Salton (1972b) cautions, however, that one cannot conclude from the above experiments that term classifications are not useful in retrieval. He suggests that strategies other than that of expanding only the queries might yield more encouraging results.

2.3.3.2 Other Methods

Gotlieb and Kumar (1968) have proposed a scheme for forming term classes based on pre-assigned semantic relationships between terms rather than on the cooccurrence of terms within documents. In their scheme, the "closeness" of two terms is determined from the sets of all immediate semantic relatives of each term. That is, if R(x) denotes the set of terms in the vocabulary either referring to x or referred to by x, then the measure of distance between two

terms, say a and b, is given by $1-[|R(a) \cap R(b)| / |R(a) \cup R(b)|]$ (note that this function measures the distance between objects). A cross-reference mapping giving the "see also" references, as is provided, for example, in the Library of Congress' Subject Headings is used to obtain the immediate semantic relatives of a term. Thus, a term-term distance matrix is obtained. The set of MCS's of the graph identified by applying a threshold to the distance matrix constitutes the set of "sharp" concepts. A scheme is also proposed for merging "sharp" concepts into larger 'diffuse' concepts. The merging of clusters is based on the following cluster-cluster distance function

$$D(C_i, C_j) = 1 - (|C_i \cap C_j| / |C_i \cup C_j|)$$

Specifically, whenever the distance between two distinct MCS's is smaller than a chosen threshold, the edges necessary to combine the vertices in the MCS's into a single MCS are inserted. This process is repeated until the distance between any two MCS's in the current graph is larger than the threshold value. The clusters generated are found to be appealing as judged by inspection and intuitive assessments.

Dattola and Murray (1967) present a method for automatically refining an existing term classification. A clustering algorithm by Rocchio (1966) is used to first cluster the document collection. Then, a set of term-term similarity matrices is obtained for each subcollection of documents. An initial set of term classes are formed using

MCS definition on the graphs associated with the various similarity matrices. Closely related initial classes are then merged and merged classes that are either duplicates or subsets of others are eliminated; the result is the set of final clusters. The authors suggest that either Tanimoto or cosine functions may be used in the formation of initial term-term similarities. In the process of merging initial clusters, however, the closeness between clusters are based on the logical overlap function.

Dattola and Murray also consider the effect of further merging of clusters based on a cluster-cluster similarity matrix. The method employed is identical to that used by Gotlieb and Kumar, except that they compute the cluster-cluster similarities based on logical overlap function. Two super-thesauri denoted THS1 and THS2, that refine the manually constructed documentation thesaurus originally available for the ADI collection included in the SMART system (Salton, 1971) data bases, are constructed. The primary difference between the thesauri is that all unique (those occurring in exactly one document) terms are grouped into a single class in THS1, while in THS2 each unique term forms a unique class by itself. For the retrieval experiments, the terms in a document or a query are replaced by the classes to which they belong and the weight of each term is divided equally between these classes. Their results for ADI collection show that THS2 performs better than THS1 which, in turn, outperforms the original manual thesaurus.

Jackson (1970, 1971) introduces an alternative scheme, which he refers to as pseudo-classification, for the generation of term classes. This scheme is based on the users' relevance assessments of the documents in relation to their queries. From a practical point of view, such an approach presumes that the system has been in operation for a period of time and that the relevance information pertaining to a number of queries has been collected during that period. Given a document collection, a term classification (which can be viewed as a term by class matrix) obtained in some way, a sample of user queries and the specification as to which of the documents are relevant to each query, he proposes that the membership of selected terms in selected classes be gradually altered by considering each query in turn. The modifications are to be made in such a way that the documents judged relevant to a query are conferred a high similarity value in relation to that query. Furthermore, the procedure is required to ensure that the alterations made to accommodate the relevance assessments of the current query do not affect the term classification in a way that would be detrimental to the queries already examined. Thus, Jackson's primary objective in proposing this scheme is to determine the limits imposed on performance by the actual design of the system. This objective stems from his suspicion that the general design of retrieval systems may be such that no choice of classification and no formulation of document-request match

function may achieve the perfect retrieval performance.

Yu (1974, 1975) has developed a methodology for the construction of term classes on the basis of the ideas proposed by Jackson. The retrieval is based on a similarity function that depends on term matches, "class matches" and "class mismatches". The term-class matrix is altered in such a way that

i) a document that is relevant to a request but does not have a sufficiently high correlation (similarity) to be retrieved is likely to score many more class matches than class mismatches so that it may subsequently be retrieved, and

ii) a document which is not relevant to a request but has a high correlation is prone to score enough class mismatches for it to be not retrieved.

Yu shows that the problem of constructing term classes according to his formulation, so that best possible performance is obtained for the sample queries, is NP-complete which implies that any algorithm that solves this problem is likely to require an exponential amount of computing time. Since the number of pairs consisting of a document and a request is usually large, this finding implies that the method is not practical. Since the method is computationally difficult, heuristics are proposed. Yu's experiments on a limited scale suggest that this method is promising from the effectiveness point of view.

2.4 Motivation Behind the Method to be Proposed for Determining Term Relationships

In all the experiments reported in the last section, the objective has been to obtain only sets of terms that are similar in the sense of being synonyms, near synonyms or referring to the same topic. However, if the premise of Jackson's and Yu's work is considered more carefully, it becomes clear that two kinds of term relationships can be hypothesized. On the one hand, if a document is relevant to a request, but the document does not have sufficiently high similarity to the request in order for the document to be retrieved, then it is likely that some of the terms used in the request, though distinct from those in the document, are similar to some terms in the document. Let the terms similar in this way be considered positively related. On the other hand, if a document is not relevant to a request but the document is retrieved, then it is probable that the terms occurring only in the document represent a context quite different from that represented by the terms that are contained only in the request. If a term is semantically dissimilar to another in this sense, then the terms are considered to be negatively related. Thus, it appears that automatic methods to identify similar or positively related terms as well as dissimilar or negatively related terms, on the basis of the relevance information provided by the users, can be developed.

Secondly, in Yu's (1974, 1975) experiments, the term classes obtained on the basis of relevance judgments corresponding to a set of sample queries are subsequently evaluated on the same set of queries. Thus, it is not known whether the term classes constructed in that study are useful in practice. Clearly, it is important that the applicability of term relationships, obtained on the basis of one set of queries, to a new set of queries be investigated.

In view of these observations, a method that permits the determination of positive as well as negative relationships between terms is proposed in the next chapter. The experiments show that when the sample queries are "representative" of the set of queries on which the term relationships are tested, improved retrieval performance is obtained. The method is shown to be efficient in that the order of computing time required is linear in the amount of input.

Chapter 3

A NEW METHOD FOR DETERMINING THE RELATIONSHIPS BETWEEN TERMS

It next will be right
To describe each particular batch:
Distinguishing those that have feathers, and bite,
From those that have whiskers, and scratch.

LEWIS CARROLL, The Hunting of the Snark

3.1 Description of the method

3.1.1 The Hypothesis

The sources of information for the method are a document collection, a set of user queries, and the relevance judgments of the documents in relation to these queries. Given this information, it is more convenient in our context to view a document request pair (or, a DR pair) as the basic unit of information. In other words, the Cartesian product of the documents with the requests constitute the collection of DR pairs. We shall assume, for the rest of the chapter, that the documents and the queries are binary vectors having a '1' or a 'Ø' in the i^{th} position depending on the presence or the absence of the i^{th} term. Before we proceed, the following definitions are introduced.

Definition 3.1.1: A DR pair satisfies assessment if either D

is relevant to R, and the measure of similarity between them is large enough for D to be retrieved, or D is not relevant to R and D is not retrieved by R. Conversely, a DR pair <u>does not satisfy assessment</u> if either D is relevant to R and D is not retrieved by R, or D is irrelevant to R and D is retrieved.

<u>Definition 3.1.2</u>: Let $D=(a_1,a_2,.....a_m)$ and $R=(b_1,b_2,.....b_m)$ be the vectors corresponding to a document and a query, where a_i <b_i> is 1 or Ø depending on whether the document <query> contains the i^{th} index term and m is the number of distinct terms in the indexing vocabulary. Then, the cosine correlation between D and R is given by the function

$$f(D,R) = \sum_{i=1}^{m} a_i.b_i \;/\; \text{SQRT}\left[\left(\sum_{i=1}^{m} a_i^2\right) \cdot \left(\sum_{i=1}^{m} b_i^2\right)\right]$$

Note that the function is a similarity (or closeness) measure. Thus, the greater the value of f(D,R), the more similar is the document D to query R.

Consider a document containing the following terms.

D_1 = {garment, fibre, summer, production}

Let R_1 and R_2 be two requests as given below.

R_1 = {shirt, cotton, production}

R_2 = {garment, fibre, winter, consumption}

Suppose that D_1 is relevant to R_1 but not to R_2. Assuming

that the retrieval is based on the cosine function, we have $f(D_1,R_1)=0.2886$, and $f(D_1,R_2)=0.5$. If a threshold value of 0.3 is chosen then neither D_1R_1 nor D_1R_2 satisfies assessment.

The above example suggests the following interpretation. Since D_1 is relevant to R_1, but not retrieved by R_1, we expect some of the terms in the set (D_1-R_1)[3] to be similar in meaning to some terms in (R_1-D_1). In this case, the terms (garment,shirt) and (fibre,cotton) may be identified as being positively related. In contrast, D_1 is not relevant to R_2, but it is retrieved by R_2. Thus, it is expected that the terms in (D_1-R_2) represent a context different from that of the terms in (R_2-D_1). In reference to the example, terms such as (winter, summer) and (production, consumption) may be considered negatively related. If we assume that the indexing process, the choice of threshold, and the users' relevance judgments are reasonable, then it is very likely that such relationships between terms exist.

In order to state the hypothesis on which our method for the determination of the relationships between terms is based, the following definitions are introduced.

Definition 3.1.3: Let $t_1,t_2,.....t_m$ be the terms used in the indexing vocabulary of the given collection. Then, the

---

[3] If A denotes a set of terms and B denotes another, then the set of terms in A but not in B are represented by (A-B).

unordered Cartesian product of (D-R) and (R-D),

$$(D\text{-}R)\&(R\text{-}D) = \{\ [t_i,t_j] \mid \text{either } t_i \in (D\text{-}R) \text{ and } t_j \in (R\text{-}D), \text{ or } t_i \in (R\text{-}D) \text{ and } t_j \in (D\text{-}R)\}.$$

Definition 3.1.4: If $[t_i,t_j]$ is in the unordered Cartesian product (D-R)&(R-D) for some DR pair, then $[t_i,t_j]$ is termed a potential term pair.

Our hypothesis is stated as follows:

Association Hypothesis: The terms of a potential term pair of a DR pair that does not satisfy assessment have a non-zero probability of being either positively or negatively related to each other. Furthermore, considering the set of DR pairs containing the term pair, the larger <smaller> the ratio of the number of DR pairs where D is relevant to but not retrieved by R to that where D is not relevant to but retrieved by R, the greater is the probability of the terms being positively <negatively> related.

The experimental work to be described in this chapter tests, essentially, this basic hypothesis. Thus, the objective of this part of the research is to determine the usefulness of the term relationships so obtained in improving the effectiveness of retrieval.

3.1.2 The Proposed Method

Given a matching function and a threshold of retrieval, each DR pair in the given set is identified as either satisfying assessment or not satisfying assessment. For each DR pair that does not satisfy assessment, the sets (D-R) and (R-D) are formed. Thus, we have many pairs of terms, which have non-zero probability of being related. The proposed method extracts from these term pairs, statistics that can be used to specify how two terms, say t_i and t_j, are related.

Given a potential term pair (t_i,t_j), let $POS(t_i,t_j)$ be the number of DR pairs such that D is relevant to but not retrieved by R and let $NEG(t_i,t_j)$ denote the number of DR pairs where D is not relevant to but retrieved by R. The measure of relationship between the terms can be determined as a function of these POS and NEG counts. The function has the following properties.

1. Given two distinct term pairs (t_1,t_2) and (t_3,t_4) for which the NEG <POS> counts are the same, but the POS <NEG> count for (t_1,t_2) is greater than that for (t_3,t_4), the positive <negative> relationship between t_1 and t_2 is likely to be stronger than that between t_3 and t_4.
2. For any given query, as there are many more documents not relevant to it than the number of relevant ones, the NEG count of a random term pair

is likely to be greater than the POS count.

Hence, the following measure is chosen.

$$\mathcal{C}(t_i,t_j) = \frac{POS(t_i,t_j) - k*NEG(t_i,t_j)}{POS(t_i,t_j) + k*NEG(t_i,t_j)} \qquad (3.1.1)$$

In the above expression, k is the ratio of POS count to NEG count in a random term pair. That is, it is expected that for a random pair of terms the positive count equals k times the negative count. By function (3.1.1), the value of $\mathcal{C}$ for such a pair is zero. Thus, the measure indicates by how much a term pair deviates from a random term pair. Since the importance of the relationships between the various terms may vary from one query to another, the value of k is made to be dependent on the query characteristics. Thus, if

$P = \{$ potential pair (t_i,t_j) | either t_i or t_j is in R$\}$

then, the value of k with respect to the query, R, is given by

$$\sum_{(t_i,t_j)\in P} POS(t_i,t_j) \;/\; \sum_{(t_i,t_j)\in P} Neg(t_i,t_j).$$

A pair of terms is considered positively related if $\mathcal{C} > 0$ and negatively related if $\mathcal{C} < 0$.

Our method computes the POS and NEG counts for all potential term pairs whose DR pairs do not satisfy assessment. The term relationships are then incorporated into the retrieval process by using a modified similarity function. The general form of similarity function suggested

is the following.

$$f'(D,R)=f(D,R) + g(D,R) \qquad (3.1.2)$$

where f(D,R) is the matching function whose value is determined only by the number of terms in common, and the number of terms not in common between D and R, and the function g(D,R) is specified in such way that it receives a positive value if the DR pair, in a rough sense, contains more term pairs that are likely to be similar than those that are likely to be dissimilar. A positive value for g(D,R) improves the chances of D being retrieved. Conversely, if R contains terms that represent a context different from that represented by the terms in D, then it is likely that the relationships assigned by the method to the term pairs in DR are mostly negative. In this case, g(D,R) will be negative and D is less likely to be retrieved. Thus, better retrieval results are expected when the measures of relationship are suitably incorporated into a similarity function of the above form.

The processing described thus far completely ignores the DR pairs that satisfy assessment. It is desirable to use these DR pairs for testing and refining the term relationships which are established using the DR pairs that do not satisfy assessment. Thus, for each DR pair that satisfies assessment, the measure of similarity is computed using the modified function in (3.1.2). If it fails to

satisfy assessment the POS and NEG counts of the various pairs of terms are adjusted as was done before. If the objective is to maximize the number of DR pairs satisfying assessment by the assignment of relationships between terms, then this refinement process can be continued until no further improvement is obtained. In our case, the procedure is terminated after all DR pairs are processed exactly once, due to economic considerations. Details of this processing is presented in appendix A.1.

3.2 The Experimental Design

3.2.1 Test Collections

The experiments have been carried out on the ADINUL (collection of 82 documents and 35 queries in the field of documentation) and the CRN4NUL (424 documents and 155 queries on aerodynamics) collections available through the SMART (Salton, 1971) system. The collections also include information for each query as to which of the documents in the collection are relevant.

3.2.2 Experimental Specifications

Our method uses relevance information concerning a given set of user queries, and the associations between terms are made in such a way that the retrieval of documents

using the modified similarity function is more likely to promote the ranks of the documents relevant to the given queries and demote the ones not relevant. Such a strategy is of practical value only if the relationships determined by using the relevance information pertaining to a given set of requests can be used to improve the retrieval performance for other queries.

To facilitate the testing of whether the relationships are applicable to other queries, the query collection is partitioned into two groups, the base set and the evaluation set. The partitioning strategy ensures that every term in the queries of the evaluation set is contained in at least one query in the base set, so that most of the relationships that will be required for the evaluation set will have been obtained from the base set. The algorithm is described in appendix A.2. This process yields 31 and 3 queries respectively in the base and evaluation sets for the ADINUL collection. For the CRN4NUL collection the corresponding numbers are 109 and 41. For each query in the base set the documents that rank in the top i positions are considered retrieved (i is 5 and 15 respectively for the ADINUL and CRN4NUL collections). The positive and the negative counts for the various potential term pairs are then obtained using the procedure proposed. The inclusion of relationships

between terms that have high document frequencies[4]. has been found to result in poor retrieval performance (Yu and Raghavan, 1977). Consequently, no associations are made between a term having high document frequency and any other term.

The retrieval is then performed for the queries in the evaluation set in two ways; first using the cosine similarity function, and then with the new similarity function which incorporates the cosine function as well as the term relationships obtained through the queries in the base set. Since the idea of making negative associations between pairs of terms on the basis of relevance information is new, it is considered imperative to show that positive as well as negative relationships contribute to the improvement in the retrieval performance. Consequently, the performance of the retrieval based on the cosine function alone is compared, in turn, to those of similarity functions that incorporate

i) the cosine function, and the positive as well as negative relationships between terms (mode 1)

ii) the cosine function, and only the negative relationships (mode 2), and

iii) the cosine function, and only the positive relationships (mode 3).

---

[4] The document frequency of a term is the number of documents in the collection that contain the term.

More specifically, the new similarity under mode 1 is the following form:

$$f'(D,R) = \text{Cosine}(D,R) * a_1 + g_1 \text{ (Positively related terms)} * a_2 + g_2 \text{ (Negatively related terms)} * a_3 \qquad (3.2.1)$$

where the a_i's, $1 \leq i \leq 3$, reflect the relative importance of the three components, and g_1 and g_2 represent the two components of g(D,R) in (3.1.2). For retrieval under the other two modes, the appropriate component is dropped from (3.2.1). Refer to appendix A.3 for further details on g_1 and g_2.

Since the cosine function represents the similarity between a document and a request in terms of the indexer supplied information, whereas the positive and negative components are determined on the basis of the user supplied information, it is assumed that both sources of information are equally important. In addition, it is assumed that the positive and the negative relationships are equally important. Thus, a_1 is taken to be 1, a_2 = Ø.5, and a_3 = Ø.5.

The standard recall[5] and precision[5] measures are used for comparing the performances of the different strategies. The overall performance of a strategy is determined by processing a number of requests with the strategy and computing the average precision over all the requests for each selected recall value. This averaging process usually requires an interpolation method since the number of relevant documents differs from one query to another. The evaluation routines incorporated into the SMART system (Salton, 1971) have been used for our experiments. These routines compute the average precision values at recall levels of 0, 0.05, ... 0.95, and 1. The percentage increase achieved in the area under the recall-precision curve for one strategy over another is considered an indicator of their relative retrieval performance.

3.3 Basic Experimental Results

3.3.1 The Effect of Weight Functions

Experiments have been carried out with a number of weighting functions. **Tables 1 and 2** summarize the for the ADINUL and CRN4NUL collections respectively under the 3 different modes, and for two different weight functions W_1 and W_2. Appendix A.4 provides specific details on the

---

[5] Recall is defined as the proportion of relevant documents retrieved and precision is the proportion of the retrieved documents actually relevant.

R	cosine function	weight function W_1			weight function W_2		
		mode 1	mode 2	mode 3	mode 1	mode 2	mode 3
Ø.1	Ø.5306	Ø.7083	Ø.7083	Ø.6392	Ø.7381	Ø.7083	Ø.7193
Ø.2	Ø.5306	Ø.5972	Ø.5972	Ø.5281	Ø.7381	Ø.7083	Ø.5526
Ø.3	Ø.5306	Ø.5972	Ø.5972	Ø.5281	Ø.7381	Ø.7083	Ø.4860
Ø.4	Ø.3408	Ø.3979	Ø.3942	Ø.3370	Ø.4333	Ø.3999	Ø.3818
Ø.5	Ø.3408	Ø.3979	Ø.3942	Ø.3370	Ø.4333	Ø.3999	Ø.3818
Ø.6	Ø.3259	Ø.3979	Ø.3942	Ø.3278	Ø.4199	Ø.3999	Ø.3439
Ø.7	Ø.3259	Ø.3979	Ø.3942	Ø.3272	Ø.3359	Ø.3999	Ø.3439
Ø.8	Ø.3259	Ø.3979	Ø.3942	Ø.3272	Ø.3359	Ø.3999	Ø.3439
Ø.9	Ø.1671	Ø.3979	Ø.2451	Ø.2927	Ø.2702	Ø.3444	Ø.3020
1.0	Ø.1671	Ø.3979	Ø.2451	Ø.2987	Ø.2702	Ø.3444	Ø.3020
average % improvement over cosine function		+31.15%	+23.37%	+11.26%	+31.63%	+33.42%	+18.19%

TABLE 1. Average precision values at ten recall points for W_1 and W_2 under 3 different modes are compared to those for the cosine function (ADINUL).

R	cosine function	weight function W_1			weight function W_2		
		mode 1	mode 2	mode 3	mode 1	mode 2	mode 3
0.1	0.6719	0.7028	0.6884	0.7011	0.7901	0.7184	0.7800
0.2	0.6048	0.6307	0.6034	0.6354	0.6957	0.6286	0.6724
0.3	0.5075	0.5331	0.5046	0.5322	0.6226	0.5384	0.5525
0.4	0.3548	0.3964	0.3669	0.3680	0.4596	0.4204	0.4111
0.5	0.3204	0.3599	0.3344	0.3345	0.4258	0.3706	0.3608
0.6	0.2667	0.2965	0.2752	0.2849	0.3593	0.3138	0.3139
0.7	0.2125	0.2472	0.2318	0.2260	0.3081	0.2821	0.2456
0.8	0.1765	0.2120	0.1967	0.1912	0.2513	0.2260	0.2109
0.9	0.1380	0.1615	0.1555	0.1466	0.1884	0.1886	0.1617
1.0	0.1342	0.1561	0.1519	0.1419	0.1813	0.1841	0.1515
average % improvement over cosine function		+8.68%	+3.78%	+4.72%	+25.43%	+13.34%	+14.28%

TABLE 2. Average precision values at ten recall points for W_1 and W_2 under 3 different modes are compared to those for the cosine function (CRN4NUL).

structure of these functions. An important distinction between W_1 and W_2 is that W_2, for the most part, places more confidence on the term relationships determined by our method than does W_1.

With W_1 on ADINUL collection the percentage improvements over using only the cosine function are 31.15, 23.37, and 11.26, respectively, under modes 1, 2, and 3. The performance when both the positive and the negative relationships are used (mode 1) is better than when either only the positive, or only the negative relationships are used. For ADINUL the improvements in retrieval over that of cosine function when W_2 is used are 31.63, 33.42 and 18.19 percent for the 3 different modes.

The corresponding results for the CRN4NUL collection are as follows. For W_1 the percentage improvement obtained for modes 1, 2 and 3 are respectively 8.68, 3.78 and 4.72. With W_2 the percentages are 25.43, 13.34 and 14.28. The results of these experiments imply that both the positive as well as the negative relationships are effective independently in providing better performance. Furthermore, the effect on the retrieval performance of positive and negative relationships seem to be additive.

With W_2 there is a <u>drastic</u> alteration of the ranks of documents. In other words, many of the documents initially not retrieved are retrieved with the new function, and vice-versa. This appears to be due to the fact that W_2 places

greater emphasis on the relationships between terms than does W_1. Although there is still substantial improvement on the average, the great fluctuations in retrieval performance from one user query to another may not, in general, be desirable. However, the emphasis on the use of term relationships (with a weighting strategy such as W_2) may retrieve documents that are not retrievable under normal conditions. Previous investigations in document retrieval indicate that some queries exhibit poor retrieval performance even when feedback strategies are employed (Rocchio and Salton, 1965; Yu et al., 1976). The reason is that the documents relevant to a query may appear in a number of isolated clusters in the document space. Usually, the feedback strategies move the query close to only one of these clusters, and it becomes difficult to retrieve documents that are in the other clusters. Under such a situation, the retrieval strategy based on term relationships, as described, may be profitable. The results with W_1 are more consistent, but the average performance is not as impressive. In this sense, these two functions are viewed as being the two extremes of a spectrum of weighting schemes. A weight function that uses additional information about the terms is devised, and it is expected that the retrieval performance of this function would represent a compromise in terms of the two extremes mentioned above.

3.3.2 The Effect of Document Frequencies of Terms

One measure of usefulness of an index term to a collection is the extent to which its presence separates the documents from one another in the indexing space (Salton et al., 1975). It has been found that neither the terms that occur in very few documents nor those occurring in very many documents are good discriminators. It is also known that high document frequency terms are unreliable for defining term relationships, and that incorporating them into the retrieval process actually results in poor performance as demonstrated in (Sparck-Jones and Jackson, 1967; Sparck-Jones, 1971; Salton, 1972a; Yu and Raghavan, 1977, 1978). In our context, it is easy to see that terms having low document frequencies are not as likely to occur as potential term pairs as are high document frequency terms. Thus, if there are two term pairs having the same POS and NEG counts, but the average document frequencies of the terms in one pair is substantially lower than that of the other pair, then higher significance should be attached to the lower document frequency pair. Weight function W_3 which takes the document frequency of terms into account is such a modification of W_1. Appendix A.5 should be referred to for further details.

Tables 3 and 4 summarize the experimental results obtained with this function for the ADINUL and CRN4NUL collections, respectively. For both these collections the

R	cosine function	weight function W_3		
		mode 1	mode 2	mode 3
0.1	0.5306	0.7083	0.7083	0.6503
0.2	0.5306	0.7083	0.7083	0.5392
0.3	0.5306	0.7083	0.7083	0.5392
0.4	0.3408	0.3991	0.3950	0.3528
0.5	0.3408	0.3991	0.3950	0.3528
0.6	0.3259	0.3991	0.3950	0.3393
0.7	0.3259	0.3991	0.3950	0.3387
0.8	0.3259	0.3991	0.3950	0.3387
0.9	0.1671	0.3991	0.2691	0.3387
1.0	0.1671	0.3991	0.2691	0.3387
average % improvement over cosine function		+35.91%	+29.16%	+15.95%

TABLE 3. Average precision at ten recall points for W_3 under 3 different modes are compared to those of the cosine function (ADINUL).

R	cosine function	weight function W_3		
		mode 1	mode 2	mode 3
0.1	0.6719	0.7419	0.6933	0.7064
0.2	0.6048	0.6509	0.6083	0.6406
0.3	0.5075	0.5446	0.5076	0.5320
0.4	0.3548	0.4147	0.3762	0.3781
0.5	0.3204	0.3734	0.3452	0.3395
0.6	0.2667	0.3233	0.2904	0.2976
0.7	0.2125	0.2592	0.2426	0.2297
0.8	0.1765	0.2161	0.2011	0.1958
0.9	0.1380	0.1636	0.1617	0.1466
1.0	0.1342	0.1570	0.1575	0.1418
average % improvement over cosine function		+13.46%	+5.87%	+6.17%

TABLE 4. Average precision at ten recall points for W_3 under 3 different modes are compared to those for the cosine function (CRN4NUL).

improvement obtained with W_3 is greater than that obtained with W_1 under each of the 3 modes, implying that document frequencies do have an impact on the significance of the term relationships. While there is not a significant difference in the performance of W_2 and W_3 for the ADINUL collection, the performance of W_3 on the average is not as good as W_2 for the CRN4NUL collection. However, in this case, the performance of W_3 is found to be more consistent than W_2, and is, therefore, an attactive alternative.

3.4 Experiments With Clustered Queries

In the experiments of the last section the queries of the collection are divided into 2 sets, and the term relationships determined by processing the queries in the base set are applied to the queries in the evaluation set. The effectiveness of this experimental strategy is dependent on the extent to which the meanings of keywords remain invariant between the two sets of queries. But from a linguistic point of view, it is natural for words to change in meaning from one context to another. Thus, it may be better to require that the term relationships to be applied to any query in the evaluation set be obtained from only those queries closely related to it. Consequently, for each query in the evaluation set, a set of queries sufficiently close to it according to some criterion of similarity is identified. The term relationships are determined on the basis of this set, and are applied to the corresponding

evaluation queries. Since only a small number of queries are processed to obtain the term relationships, the number of DR pairs processed for any given evaluation query is considerably smaller than the number of DR pairs in the base set. This implies that the number of term pair relationships that would have to be kept in store at any time is much smaller for this strategy, and is therefore more economical.

The measure of similarity chosen for selecting the set of queries related to an evaluation query is given by the ratio of the number of terms of low document frequency (as characterised in appendix A.5) in common to the number of low frequency terms in the evaluation query. A threshold value is used to determine if a query is sufficiently close to the evaluation query. The experimental results for the CRN4NUL collection using the weight function W_3 for two different threshold values, 0.5 and 0.25 Are presented in **Table 5.** At a threshold of 0.5, the percentage improvements over the use of cosine function alone are 15.79, 6.87, and 12.31, respectively, for modes 1, 2, and 3. At the lower threshold, the corresponding percentages are 20.13, 8.89, and 10.93. The performance is somewhat better when a lower threshold is chosen. This trend, however, is not consistent as evidenced by a smaller gain achieved under mode 3 at the lower threshold. The query clustering strategy achieves consistently better improvement in retrieval performance compared to the query partitioning approach (refer to Tables 4 and 5) implying that the former approach is a more

R	cosine function	Each base query contains at least 50% of non-high freq. terms in test query			Each base query contains at least 25% of non-high freq. terms in test query		
		mode 1	mode 2	mode 3	mode 1	mode 2	mode 3
0.1	0.6719	0.7359	0.7067	0.7301	0.7393	0.6973	0.7288
0.2	0.6048	0.6380	0.6122	0.6527	0.6613	0.6171	0.6550
0.3	0.5075	0.5727	0.5319	0.5593	0.5930	0.5389	0.5580
0.4	0.3548	0.4597	0.3997	0.4155	0.4716	0.3977	0.4126
0.5	0.3204	0.3899	0.3476	0.3695	0.4175	0.3555	0.3649
0.6	0.2667	0.3399	0.3053	0.3150	0.3676	0.3107	0.3118
0.7	0.2125	0.2634	0.2315	0.2519	0.2801	0.2529	0.2469
0.8	0.1765	0.2283	0.1928	0.2180	0.2394	0.2090	0.2107
0.9	0.1380	0.1767	0.1533	0.1661	0.1782	0.1696	0.1475
1.0	0.1342	0.1678	0.1506	0.1603	0.1728	0.1668	0.1415
average % improvement over cosine function		+15.79%	+6.87%	+12.31%	+20.13%	+8.89%	+10.96%

TABLE 5. Average precision at ten recall points, when query clustering strategy based on simple matching function at two cutoff levels is used with W_3 under 3 different modes, are compared to those for the cosine function (CRN4NUL).

effective way of choosing base sets that are representative of the test queries.

The query clustering strategy has also been tested using a different selection criterion for the choice of the related queries. Each term is assigned a weight depending on its document frequency. The smaller the document frequency, the greater is the weight assigned. The measure of correlation is given by the sum of the weights of the terms in common. The three nearest neighbour (3-NN) criterion is used, and the set of related queries are identified. The results are presented in Table 6. The improvements in performance over the cosine function are 14.17, 6.64, and 9.12 percent under the 3 modes. These results are fairly good, and suggest that the term relationships obtained are not too sensitive to the changes in cluster formation.

3.5 The Determination of the Coefficients in the Similarity Function

3.5.1 Motivation

The modified similarity function (3.2.1) is of the form

$$f'(D,R) = a_1 * \text{cosine component} + a_2 * \text{positive component} + a_3 * \text{negative component} \qquad (3.5.1)$$

where the a_i's reflect the relative importance of the different components. When these coefficients are viewed as experimental parameters, a potentially large number of

R	cosine function	3-NN of test query based on weights of common terms form the base set		
		mode 1	mode 2	mode 3
0.1	0.6719	0.7419	0.7022	0.7065
0.2	0.6048	0.6483	0.6059	0.6432
0.3	0.5075	0.5289	0.5083	0.5495
0.4	0.3548	0.4204	0.3909	0.4002
0.5	0.3204	0.3873	0.3511	0.3563
0.6	0.2667	0.3385	0.3005	0.3146
0.7	0.2125	0.2563	0.2406	0.2348
0.8	0.1765	0.2187	0.1986	0.2012
0.9	0.1380	0.1721	0.1595	0.1524
1.0	0.1342	0.1625	0.1569	0.1443
average % improvement over cosine function		+14.17%	+6.64 %	+9.12%

TABLE 6. Average precision at ten recall points, when query clustering strategy based on a nearest neighbour criterion is used with W_3 under the 3 modes, are compared to those for the cosine function (CRN4NUL).

reasonable combinations exist for the choice of parametric values and the experimental process becomes rather tedious. Thus, it is considered desirable to investigate the possibility of determining these coefficients by analysing the information derived from the DR pairs in the base set.

3.5.2 Problem Specification

Consider the set of document - request pairs formed by pairing each query in the base set against the documents in the collection. Let DR_{ij}, $1 \leq j \leq 2$, and $1 \leq i \leq n$, denote the i^{th} DR pair in j^{th} category, where

$$j = \begin{cases} 1 \text{ if, for a DR pair, D is relevant to R} \\ 2, \text{ otherwise} \end{cases}$$

and n_j is the number of DR pairs in category j. Suppose the DR pairs are processed according to the method proposed and the POS and NEG counts are determined for the various pairs of terms. Further, suppose that the cosine, the positive, and the negative components are computed for each DR pair as before. Then, the problem is to identify coefficients to be used in (3.5.1) in such a way that the similarity measures obtained can be used to discriminate between the two categories of DR pairs in an 'optimum' way.

3.5.3 Approach

A well known multivariate data analysis method referred

to as discriminant analysis (Cooley and Lohnes, 1971) is used to derive the coefficients desired. This method has been applied to information retrieval problems before (Anderson, 1958; Williams, 1965; Chan, 1973). In these cases the method is found to offer a solution to the problem of selecting measurable attributes to represent document categories. The application of discriminant analysis to the problem of determining the importance of term-term relationships seems to be new.

Let the (column) vector X_{ij} (a 3 -tuple) denote the values of cosine, positive, and negative components for the i^{th} DR pair in the j^{th} category. X_{ij} is termed the vector variable corresponding to DR_{ij}. From a theoretical point of view, the method requires that the variable vectors have (multivariate) normal distributions within each category, and that the variance-covariance matrices of the two groups be equal. But, in practice, the method is known to be very robust (Nie et al., 1975) and these requirements need not be adhered to strongly.

Given any coefficient (column) vector A the scalars $A'X_{i1}$ and $A'X_{i2}$, $1 \leq i \leq n$, represent samples from two univariate normal distributions as shown in **Figure 2.** In the figure, m is the mean of the variable Vectors corresponding to all DR pairs and m_j the mean vector of the variates in group j. Let $d_j^A = |m_j - m|$, and let Δ_j^A be the sum of the squared deviations of the variable vectors in group j from

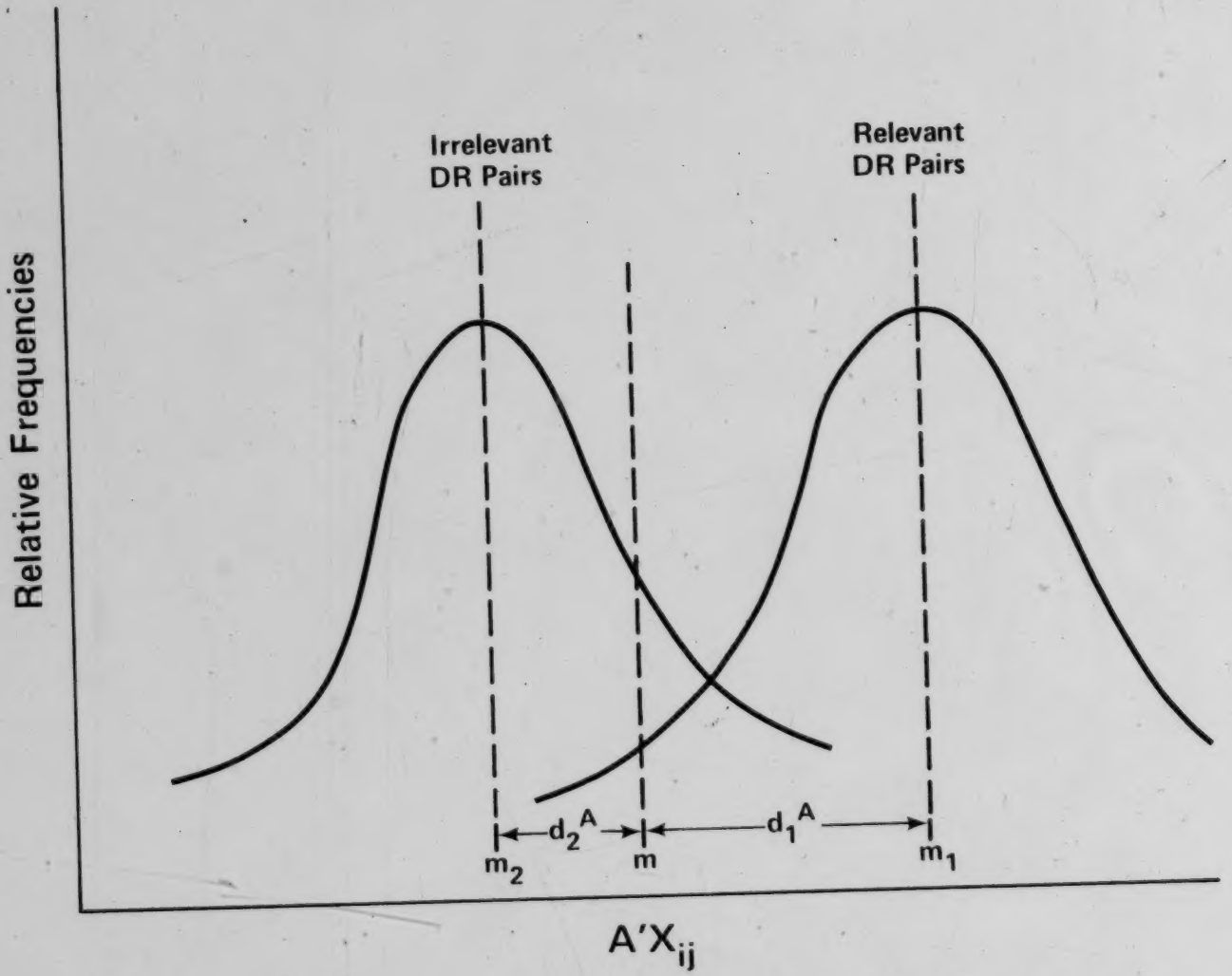


FIGURE 2. The distributions of A'X for DR pairs in which D is relevant to R, and where D is not relevant to R.

their mean, m_j. Clearly, Δ_j^A is a measure of within category variance, and d_j^A, j=1,2, are indicators of the spread between the two groups. Since our interest is to have a small value of within group variance, and a large spread between the groups, the following measure is a reasonable indicator of our requirements.

$$\theta = \sum_{i=1}^{2} n_j . d_j^A \; / \; \sum_{i=1}^{2} \Delta_j^A$$

Discriminant analysis can be used to identify an A which maximizes θ.

The variables in X are usually normalized so that the a_i's in A can be interpreted as indicating the importance of one component relative to another, and this normalization can be accomplished by dividing the variables in X by their standard deviations (Morrison, 1974). Consequently, only the values of the standardized coefficients are reported for the experiments presented in the next section.

3.5.4 Experimental Results

The CRN4NUL collection, the weight function W_1, and the query partitioning strategy are used in these experiments. The coefficient vector is obtained based on the analysis[6] of the information pertaining to the base set. The standardized

---

[6] The discriminant analysis routines developed by Nie, et. al. (1975) have been used for these experiments.

coefficients obtained are 0.4248, 0.8637 and 0.2613, respectively, for cosine, positive and negative components. The values indicate that the positive component is more than twice as important as the cosine in discriminating between the two categories. The use of these coefficients in the retrieval process results in a deterioration of about 27 percent in performance compared to the case where only the cosine function is used. It appears that the coefficients that are the best for the base set tend to overemphasize the importance of the term relationships, and are not appropriate for the evaluation set.

It has been pointed out in (Morrison, 1974) that in cases where the samples from the two groups vary greatly in size, the discriminant analysis technique puts most of the items into the larger group in order to achieve the maximum percentage of correct classifications. In our situation, there are many more DR pairs in which the document is not relevant to the request than the ones in which D is relevant. Therefore, a strategy that equalizes the sizes of samples from the two categories of DR pairs is considered. Two independent samples (of sizes approximately equal to the number of DR pairs in which D is relevant to R) are randomly selected from the set of DR pairs in which D is not relevant to R, say, sample 1 and sample 2. Two sets of discriminant coefficients are derived, first by analysing sample 1 with the smaller group (the set of DR pairs in which D is relevant to R), and then by considering sample 2 with the

same group. The standardized coefficients obtained in these two cases are given in **Table 7.** These values suggest that the cosine component is the most discriminating among the three variables being considered.

The retrieval of documents for the evaluation queries is carried out by using the average value of the corresponding coefficients obtained from the two runs. The retrieval results for this strategy are included in Table 9. The average improvement over the cosine function is 12.38%. Although the average performance is fairly good, this result is not completely satisfactory in the sense that large fluctuations in the ranks of documents (relative to the ranks for cosine function) are observed, and a number of queries in the evaluation set are found to deteriorate in retrieval performance.

In order to compensate for the effect of such fluctuations, another approach is considered. In this approach, the cosine component is awarded the same importance as that of positive and negative components combined. The motivation comes from the fact that the cosine component reflects the indexers' assessment of the relevance of a document to a request, where as the other two components are based on the relevance information provided by the user. Discriminant analysis is performed on sample 1 and sample 2 taken separately with the smaller group; but only the positive and the negative components are considered

	Cosine	Positive	Negative
sample 1 and small group	Ø.6719	Ø.5532	Ø.2788
sample2 and small group	Ø.6693	Ø.5601	Ø.2591

TABLE 7. Standardized coefficients for two runs using all three components in discriminant analysis (For each run, the sizes of samples from the two groups are about equal).

	Positive	Negative
sample 1 and small group	Ø.8618	Ø.2746
sample 2 and small group	Ø.8565	Ø.29

TABLE 8. Standardized coefficients for two runs using only the positive and the negative components in discriminant analysis (For each run, the sizes of samples from the two groups are about equal).

R	Cosine Function	Weight function W_1, mode 1		
		a_1=1.0 a_2=0.5 a_3=0.5*	Discriminant analysis, using equal size samples.	
			all a_i's are computed.	only a_2 & a_3 are computed.
0.1	0.6719	0.7028	0.7260	0.7020
0.2	0.6048	0.6307	0.6429	0.6324
0.3	0.5075	0.5331	0.5486	0.5334
0.4	0.3548	0.3964	0.4185	0.3974
0.5	0.3204	0.3599	0.3781	0.3604
0.6	0.2667	0.2965	0.3198	0.2968
0.7	0.2125	0.2472	0.2595	0.2424
0.8	0.1765	0.2120	0.2095	0.2085
0.9	0.1380	0.1615	0.1674	0.1611
1.0	0.1342	0.1561	0.1549	0.1522
average % improvement over cosine function		+8.68%	+12.376%	+8.53%

TABLE 9. Average precision at ten recall points under model for W_1, when i) all a_i's and ii) only a_2, a_3 are obtained using discriminant analysis (after the sizes of samples are equalized), are compared to those for cosine function (CRN4NUL).

* These precision values are the same as in TABLE 2

as the discriminating variables. The standardized coefficients resulting from this process are presented in Table 8. The retrieval is performed by using the average of the corresponding coefficient from the analysis described above, and by setting the coefficients for the cosine component equal to the sum of the other two coefficients. The results of this strategy are also included in Table 9. The average improvement is 8.53 percent, and this performance is comparable to that obtained with the strategy in which the coefficients a_i, $1 \leq i \leq 3$, are chosen as described in section 3.2.2.

3.6 Statistical Significance of the Experimental Results

Significance tests have been carried out for all the experiments reported here. For each strategy precision values are computed at a number of recall points and, in addition, 4 global measures of effectiveness are determined. The comparison of each new strategy against that of using cosine function alone is based on these measures. A particular strategy is considered significantly better than another if for the majority of the measures employed the probability of the two methods being equally good is less than 0.05. Some additional details concerning the significance tests are included in appendix A.6.

For the Cranfield collection of 400 documents and 155 queries (CRN4NUL), the improvement obtained in retrieval

performance over that of using cosine function alone is found to be significant in all but one experiment. The strategy of using discriminant analysis to determine all three coefficients, a_i's, of the modified similarity function and then carrying out a retrieval using the weight function W_1 is not significantly better than the retrieval based on cosine function alone (reported in **Table 9**). It seems that the poorer performance is due to the fact that the importance attached to the cosine component is smaller than the cumulative weight attached to the other two components.

In the case of the ADINUL collection of 82 documents and 35 queries, the improvement in performance obtained for the various strategies tested are not statistically significant (**Tables 1 and 3**). That is, at the chosen significance level the null hypothesis that the various strategies are no better than the use of cosine function alone cannot be rejected. However, only 3 queries could be included in the evaluation set on which the term relationships are tested. Moreover, the average improvements obtained in the precision values are in the range of 15 to 35 percent. (An improvement of over 5 percent has quite often been considered signficant in other studies.) In view of these observations, the results on the ADINUL collection are considered inconclusive, at worst. It is likely, however, that statistically significant improvements would be obtained if more queries, sufficiently similar to base

set of queries, can included in the evaluation set.

3.7 Computing Time

Let DR_i , $1 \leq i \leq N$, be the N document request pairs in the base set and S_i the number of potential term pairs in the i^{th} DR pair, given by

$$S_i = | (D-R)_i | * | (R-D)_i |.$$

The main component of the classification process is the updating of the frequencies of potential term pairs. In doing this, all DR pairs of that satisfy assessment, and some DR pairs that do not are processed. Therefore, the computing time is proportional to $\sum_{i=1}^{N} S_i$. That is, the time is bounded by

$$T_C \leq k_1 * \sum_{i=1}^{N} S_i$$

for some constant k_1. It is also worth pointing out that, for a reasonable choice of the threshold, quite a large number of DR pairs are classified as not relevant and not retrieved. Consequently, the proportion of all the DR pairs (N in total) processed is, usually, very small. Thus, k_1 is likely to be very small.

A similar result is obtained for the evaluation stage. If N' and S_i' denote, respectively, the number of DR pairs in the evaluation set and the number of potential term pairs in the i^{th} DR pair of this set, then the time bound, T_E for

the evaluation stage is given by

$$T_E \leq k_2 * \sum_{i=1}^{N} S_i'$$

where k_2 is some constant.

Chapter 4

METHODS FOR EVALUATING THE STABILITY OF CLASSIFICATION STRATEGIES

4.1 The Need for Stable Classifications

In any practical application of clustering techniques, errors are liable to occur during the collection and compilation of the information about the objects to be classified. It is, therefore, of interest to examine the effect of such errors on classifications subsequently produced. As pointed out in Chapter 1, classifications are considered stable if small changes in the input data lead only to small changes in the classification.

From the point of view of document retrieval systems, the following types of errors and problem situations are recognized.

a) In the large collections of information to which automatic classification procedures must be applicable, there is a certain probability of finding transcription errors. These errors are purely clerical in origin and arise when information is transcribed into a form suitable for processing by computer. Some of the errors may be detectable if certain conventions are imposed in the preparation of data. For instance, if the attributes are always presented in increasing order of their code numbers then a sequence

check can be performed.

b) A document may be mistakenly described by an attribute which it does not possess.

c) The environment of a document retrieval system is usually dynamic. This aspect has an impact on the completeness of the document descriptions. For example, in computer literature, the distinction now drawn between "parallel" and "serial" computation may not have been made originally, although in retrospect, the distinction would have been a relevant one. Thus, as a collection expands to include material not envisaged at the time the original documents were indexed, the indexing vocabulary also grows. Changes in the representation of documents are also brought by changes in the usage, and sometimes in the meaning, of terms over time. Consequently, document as well as term classifications would require alterations. It is desirable, however, that if document representations are only slightly incomplete or if only a few updates take place then the classifications need only minor alterations.

d) A document may not have a unique index description. That is, although two indexers indexing the same documents may produce descriptions which agree with each other substantially, it is nevertheless likely that there will be points of disagreement. Again, it is necessary that the two classifications derived from the same document collection, indexed separately by two different indexers, do not differ

substantially.

4.2 Difficulties in the Evaluation of Stability

Before it can be established that a given classification strategy is stable, techniques must be devised for the measurement of changes in the data and in the classification.

Small changes in a document-term array may be measured quite simply. Suppose that two document arrays are given, representing differing descriptions of the same document collection. There is no difficulty in identifying a particular document in one array with the corresponding document in the other array. Thus, over the whole collection, a measure of difference between the two arrays may be obtained by computing certain quantities such as the Kendall (1945) coefficient of agreement.

Changes in the classifications are less easily measurable, although a classification, like the document collection, can be represented by means of a binary array. In a classification array C, the element C_{ij} of the i^{th} row indicates the presence or absence of object j in class i. Suppose that C and C' are two classification arrays produced from two document arrays, D and D', which are differing representations of the same document collection. Since names or labels given to each class in C and C' are entirely arbitrary, it is not possible (as could be done for document

arrays) to identify each class of C as corresponding to a specific class in C'. This means, more complicated approaches must be used.

4.3 Comparison of Classifications

One of the first suggestions put forward for the comparison of different classifications was by Sokal and Rohlf (1962). They propose a method for comparing hierarchical clusters which are represented by dendrograms (diagrams of relationships). Suppose that the object-object similarity values are given and that the clustering method is such that, given any two threshold values, each cluster corresponding to a particular threshold is a subset (though, not necessarily a proper subset) of some cluster that would be created for a smaller threshold. Then, the sequence of classifications that correspond to the various possible threshold values can be summarized in a tree-like diagram. For example, consider the object-object similarity values given below corresponding to a set of 5 objects.

O_2	0.6			
O_3	0.4	0.8		
O_4	0.1	0.5	0.7	
O_5	0.1	0.2	0.2	0.3
	O_1	O_2	O_3	O_4

Suppose that the clusters corresponding to a given threshold are defined as the CC's of the associated graph. Then, the dendrogram for this situation is as shown in Figure 3. In a dendrogram the abscissa has no particular meaning. The ordinate, on the other hand, represents similarity values. The purpose of the diagram is to show the (threshold) level at which two or more objects combine to form a common cluster. In the example given, O_2 and O_3 join at level 0.8, O_4 combines with O_2 and O_3 at level 0.7, O_1 combines with O_2, O_3 and O_4 at level 0.6 and, finally, all the objects form a single cluster at level 0.3.

A summary of the level at which the various pairs of objects join is obtained as follows:

The range of similarity values along the vertical axis is divided into a suitable number of equal intervals. Suppose that the number of intervals is N and the similarity values are in the range between 0 and 1. Let the code number of the interval $((i-1)/N, i/N)$ be i for $1 \leq i \leq N$. Then, the cophenetic value of two given objects is defined to be the code number of the interval that contains the similarity value at which the objects join in the dendrogram. Using this scheme, a matrix representing the cophenetic value for every pair of objects is generated.

In our example the code numbers go from 1 to 4. The cophenetic value of the objects O_2 and O_4, for instance, is 3.

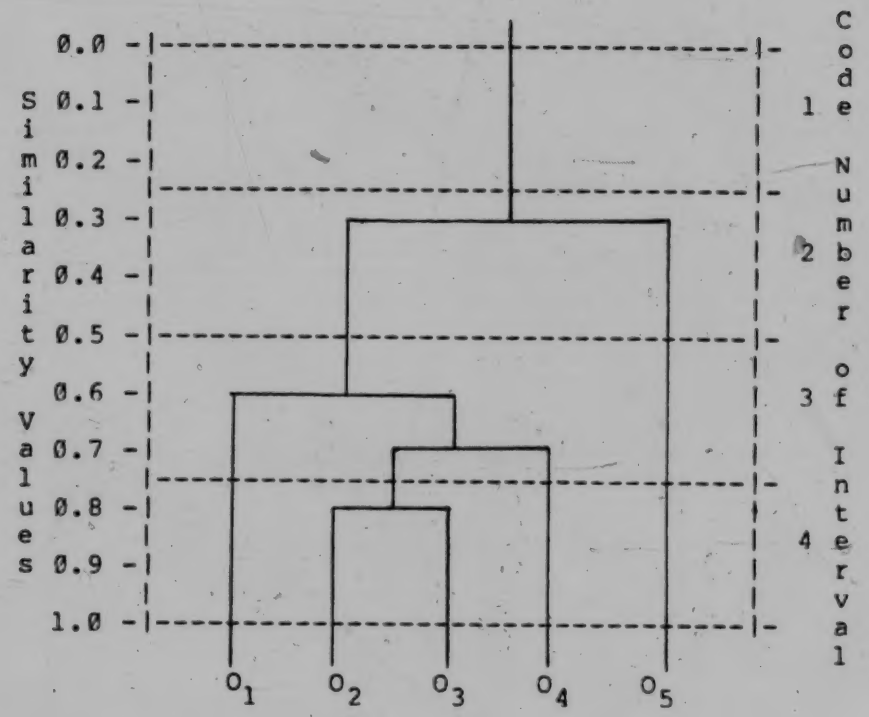


FIGURE 3. A dendrogram to illustrate the computation of cophenetic values.

Thus, given two classifications, two matrices of cophenetic values are generated and a measure of agreement (or disagreement) between them is determined by calculating an ordinary product moment correlation coefficient between the corresponding elements of the two matrices.

Jackson (1969b) has devised a scheme that measures the extent to which the object-object similarities that one can infer from the classification generated reflect the original object-object similarities from which the classification is obtained. This measure can be considered an example of intrinsic criteria referred to in Chapter 1. Thus, he is only interested in comparing different classifications generated from the same input data, but using different classification methods. Jackson's scheme is, however, described here as it appears to be applicable also to our context.

Let D denote the object-attribute binary array and C the object-class binary array. Let S and T denote the object-object similarity matrices obtained respectively from D and C by applying a similarity function to all the pairs of object descriptions. Then, the assessment of the discrepancy (or agreement) between S and T is made by checking whether or not sign (S(i,j)-S(k,l))=sign(T(i,j)-T(k,l)) for each distinct i,j,k and l that represent four different objects.

In other words, the greater the number of cases

(distinct quadruplets) in which this condition holds, the better is a classification. Although a simple ratio of the number of discrepancies to the total number of quadruplets may be used, it is suggested that a better understanding as to the goodness of the classification will be obtained if such a proportion is computed first for the most strongly related pairs, then for less similar pairs and so on to the least similar ones. This process will require the application of a number of thresholds to matrices S and T and the computation of the proportion of discrepancy separately for each threshold. In a situation where two classification arrays obtained from two differring descriptions of the same document collection are to be compared, the classification arrays would simply replace the arrays C and D described above.

In a recent study Yu (1976) proposes a method for measuring the amount of disturbance in classification due to small changes in the input data. The measure is developed in particular reference to graph theoretical clustering strategies. The changes are assumed to be small enough that the objects are not modified drastically. Consequently, the changes in the graph consist only of addition and deletion of edges.

Given the initial and the perturbed graph and a classification scheme, the amount of disturbance is measured by the minimum number of edge modifications required to

correct the perturbed graph so that the clusters of the initial and the corrected graphs are identical according to the cluster definition given. Yu finds that clusters defined as the MCS's require the maximum number of operations. In fact, in this case, the corrected graph must be identical to the initial graph. He compares the effect of simple matching and cosine similarity functions experimentally and concludes that for both cluster defining methods tested (CC and MCS), clusters produced on the basis of cosine function are less stable than those of simple matching.

Corneil and Woodward (1978) have compared the methods of Vaswani, Gotlieb and Kumar (both described in Chapter 2), and Zahn (1971) (which is based on the maximal spanning tree of the graph obtained from object-object similarities) with respect to a number of cluster properties. One of the properties considered is stability. They measure the difference, between classifications obtained from perturbed and unperturbed data, in terms of the following quantities:

N_s = the number of similarly clustered objects; that is, the number of pairs of objects either in the same cluster in both the classifications or in different clusters in both the classifications.

N_d = the number of dissimilarly clustered objects; that is, the number of pairs of objects in which the pair is in the same cluster in one classification and in different clusters in the other.

In comparing different clustering methods, the one least affected by perturbations would maximize N_s and minimize N_d. Their experiments with 5 sets of randomly generated points (having varying amounts of separation between points and leading to graphs with varying amounts of edge density) show that

(a) the stability of Zahn's method is independent of edge density but becomes more stable, from being worse than the other two methods to being much better, as the separation is increased.

(b) Vaswani's and Gotlieb and Kumar's method are about equally stable as density and separation are varied. When the separation is not high, these two methods are found to be independent of edge density. In the case of the largest separation tested, both methods become more stable as the edge density increases.

A work related to the problem of stability, but not directly concerned with the comparison of classifications is due to Jackson (1969a, 1972). He assumes that the document descriptions are binary and errors are independent and equiprobable. Let f_t and f_u respectively be the values of the similarity function f between two given objects before and after some entries of the description have been altered. Then, his aim is to compute the statistical estimate, $\emptyset^*$, of some function, $\emptyset$ (say, the mean square deviation), of f_t and f_u. This estimate is given by

$$\emptyset^* = \overline{\sum_{ses^*}} P(S) \cdot \emptyset (f_t, f_u)$$

where S is a particular change in the object descriptions, S* is the set of all possible changes and P(S) is the probability of occurrence of S. Jackson's effort is mainly involved with the development of an efficient algorithm to compute Ø*.

4.4 Motivation for the Approach Selected for Stability Analysis

The review of earlier work in the comparison of classifications suggests that there is a considerable amount of choice in the measure that might be selected. What is common to most of these approaches is that a measure of discrepancy between classifications is devised, two classifications, one from unperturbed and the other from perturbed input data, are obtained and the stability of the classification strategy is assessed in terms of the proposed measure. Thus, the earlier studies on the problem of stability have been experimental.

It is found, however, that the measure proposed by Yu can be used to evaluate graph theoretic clustering strategies in a formal setting. Since graph theoretic strategies are predominant in the construction of term classes (see Chapter 2), it is desirable to compare the stability of such techniques analytically. The details of the analysis and the results obtained are presented in the next chapter.

Chapter 5

STABILITY ANALYSIS OF CERTAIN GRAPH THEORETIC CLUSTERING METHODS

5.1 Introductory Concepts Relating to the Measure of Stability

5.1.1 Preliminary Definitions on Graphs

The idea of using graphs to represent the closeness between the objects to be classified was introduced in Chapter 2. In this section, some related facts are presented.

A graph, in our context, consists of vertices that represent the objects and an edge is associated with a pair of vertices if the measure of closeness between the corresponding objects is sufficiently large (as determined by a threshold). Thus, when two vertices are adjacent (have an edge between) the associated objects are very close to each other.

An MCS of such a graph represents a subset of the objects in which each object is closely related to every other object in the subset. Since all possible edge

connections are made, a MCS in n vertices has C(n,2)[7] edges. On the other hand, a CC represents a subset of objects that are related to, but not necessarily very close to each other. That is, two objects that are in the same CC but not adjacent are only connected via other objects. Next, we consider some properties associated with CC.

Definition 5.1.1: Let P be a property. A minimal set S' having property P is a set such that S' - {x} does not have P for any x $\in$ S'.

Definition 5.1.2: A CC, $G_1 = (V_1, E_1)$, of a graph is minimally connected if it is minimal with respect to the edge set E_1.

Thus, a minimally connected graph does not have any edge that can be removed without causing the resulting graph to be disconnected. It is easy to show by induction that such a graph in n vertices has (n-1) edges and has no cycles, where a cycle is an alternating sequence of vertices and edges in which the first and the last vertices are identical. A related fact is that any edge in a cycle can be removed without affecting the connectivity of the graph.

If v and w are adjacent and if the deletion of this edge would cause v and w to be disconnected, then the edge

[7] C(n,2) = n(n-1)/2, the number of 2 combinations of n elements.

(v,w) is a bridge.

We next introduce the notion of a complement graph.

Definition 5.1.3: $\overline{G} = (V,\overline{E})$ is the complement graph of $G = (V,E)$ if $\overline{E} = \{e \mid e \in (E'-E)\}$, where E' is the set of edges in the complete graph on the vertex set V.

5.1.2 Notations Relating to the Measure of Stability

The measure suggested by Yu (1976) estimates the amount of change in a set of clusters by the minimum number of 'operations' required to restore the set of modified clusters to the original ones, where an operation is either an addition or a deletion of an edge. More precisely, let $G = (V,E)$ be the graph that would represent the object-object similarities if there had been no errors in the input data. Let $G^* = (V,E^*)$ denote the graph actually obtained as the result of some perturbations in the input. That is, E^* is obtained possibly by deleting some edges from E and adding some edges from $\overline{E}$ to E. Thus, edge deletions come from the original graph, whereas edge additions are from the complement graph. Given a cluster defining method D, suppose $G^{**} = (V, E^{**})$ denotes a graph which is obtained through minimum number of changes to G^* such that G and G^{**} have an identical set of clusters. Then, the amount of change is specified by the expression $|(E^{**}-E^*) \cup (E^*-E^{**})|$. This measure of the amount of work is denoted W(D).

As there are a great number of different graph theoretic cluster definitions in use, it is not possible to compute this quantity for every definition of a cluster. However, certain well known cluster definitions are analysed in terms of how their characteristics relate to this measure of stability.

5.2 Stability Analysis of Certain Cluster Defining Methods

5.2.1 Cluster Characteristics

It has been shown by Yu (1976) that the amount of 'work' required to convert a set of modified clusters to the original ones is as much as it can possibly be when clusters are defined as the MCS's. For the sake of completeness, the following proposition is presented without the proof.

Proposition 5.2.1: Let $\{c_1, \ldots, c_i\}$ and $\{c_1^{**}, \ldots, c_j^{**}\}$ be two sets of clusters defined as the MCS's, respectively, of the graphs $G = (V,E)$ and $G^{**} = (V,E^{**})$. Then $\{c_1, c_2, \ldots, c_i\} = \{c_1^{**}, \ldots, c_j^{**}\} \Longrightarrow G = G^{**}$.

The proposition says that the only way to convert from one set of clusters to another set is to make the two sets of edges identical. Thus, the amount of work to go from the graph G^* to G^{**} is given by $|(E - E^*) \cup (E^* - E)|$.

We now develop certain ideas which enable us to assess

the stability of certain cluster definitions relative to that of the clusters obtainable using the CC definition. The basic objective of clustering is to obtain groups of closely related objects. Conversely, it is reasonable to ensure that only related objects are placed in the same cluster. Since, in a graph, the connectedness of two vertices represents the fact that the corresponding objects are related in some way, the following condition is imposed on the clusters that are obtained.

Definition 5.2.1: A cluster is unfragmented if for any two vertices v and w in the same cluster, v and w are connected.

This property implies that any cluster must be a subset (though, not necessarily a proper subset) of a connected component. In the rest of the chapter the term cluster will refer only to unfragmented clusters.

Just as we place unrelated objects in different clusters, it is also quite natural to always have close objects placed in some cluster. This notion is formalized in the following way.

Definition 5.2.2: A cluster definition is adjacency oriented if for every pair of adjacent vertices v and w in a graph, the application of the definition to the graph yields at least one cluster that contains both v and w.

It is easy to see that clusters defined as the CC's and the MCS's are adjacency oriented and unfragmented. One of

the basic difficulties with the CC definition is the well known "chain effect" (Wishart, 1969; Jardine and Sibson, 1971). That is, there may exist a sequence of n connected objects where each object is adjacent to at most its two neighbors. In this case, the n objects are placed in a cluster, but the two objects at the extremes of the chain (sequence) may have very little in common. On the other hand, an MCS requires the strict condition that every object in a cluster be adjacent to every other object in the cluster.

Consequently, cluster definitions that represent a compromise between these two extremes have been considered more appropriate for many applications. Many such strategies have been investigated in earlier studies (Bonner, 1964; Abraham, 1965b; Dattola and Murray, 1967; Sparck-Jones, 1967, 1971; Gotlieb and Kumar, 1968; Jardine and Sibson, 1968a,b; Vaswani, 1968; Auguston and Minker, 1970a; Sibson, 1970; Ling, 1972; Koontz, et al., 1976; Day, 1977; Matula, 1977). Several of these cluster defining methods are adjacency oriented. They are described next as they are relevant to subsequent discussions.

Bonner (1964) and Needham (1967), among others, have suggested that the recognition of MCS's should be the first step in the process of identifying clusters. Accordingly, the set of MCS's corresponding to the graph obtained after the application of a threshold to the similarity matrix have

been considered to be the 'nucleus' of the desired classification by Gotlieb and Kumar, Jardine and Sibson, and others. These classification schemes, in general, involve the merging together of the MCS's in the original graph that are sufficiently close, by some criterion, to each other. Let such clustering methods be called MCS centered. Since the MCS's corresponding to the original graph are adjacency oriented and since each final cluster is either identical to one of the MCS's or is the union of some of them, it follows that MCS centered methods are adjacency oriented.

The method of Gotlieb and Kumar (also used by Dattola and Murray, and Auguston and Minker) has already been described. Jardine and Sibson have proposed different types of restrictions that can be placed on the degree of overlap allowed between distinct MCS's. In their methods, the merging of the MCS's is performed on the basis of such restrictions. Jardine and Sibson call their methods fine k-clustering (B_k); Coarse k-clustering (B_k^c) and u-diametric clustering (C_u) where $k \geq 1$ and u is a constant in the range $[0,1]$.

The sequence of fine k-clustering methods, B_k, are given a simple graph theoretic description that generalizes the CC definition. The procedure for obtaining the B_k clusters is as follows:

Given a similarity (or dissimilarity) function and a threshold level, the corresponding graph is generated. The

MCS's are marked, and whenever there are k or more vertices in common between two such subgraphs edges are inserted to make the union of the vertices in the two subgraphs into a single complete subgraph. This process is repeated until there is no further alteration.

It is clear that at the end of the process the overlap between any two distinct MCS's in the new graph is no more than (k-1). Thus, we have B_1=CC and if there are altogehter p vertices in the graph, then B_{p-1}=MCS. Coarse k-clustering methods are quite closely related to the methods B_k. As they do not have quite the simple interpretation in terms of the overlap between MCS's, they are not described here. The methods C_u ensure that, for given u, the diameter of the set of objects in the overlap of any two MCS's is less than or equal to u.h, where h is the threshold level at which the initial graph is obtained.

A somewhat different approach may be taken to identify cluster defining methods that represent a compromise between CC and MCS. More specifically, in view of the fact that it is sufficient to have n-1 edges to form a CC in n objects whereas C(n,2) edges are needed to form a complete subgraph, the following generalization is suggested.

Definition 5.2.3: A subgraph $G_1 = (V_1, E_1)$ of G, where $|V_1| = p$, is t-complete if it is connected and has at least

$\lceil t.C(p,2) \rceil$[8] edges, $0 \leq t \leq 1$.

We wish to propose a t-complete subgraph as a reasonable way to specify a cluster. In order to avoid the possibility of an identified cluster being a subset of another, it is ensured that a cluster is enlarged until it is not possible to increase its size without losing the property of t-completeness. This requirement leads to the following definition.

Definition 5.2.4: A subgraph $G_1 = (V_1, E_1)$ is maximal t-complete if it is t-complete and is maximal with respect to V_1.

Let D_t denote the definition whose clusters are given by all maximal t-complete subgraphs of G. Clearly, D_1 = MCS and D_0 = CC. Thus, D_t, for $0 < t < 1$, represents a compromise between the clusters defined as the CC's and those defined as the MCS's in terms of the cluster properties. It is easy to see that this family of definitions are also adjacency oriented and produce clusters that are unfragmented.

In the case of cluster definitions that are not adjacency oriented, there may be adjacent vertices that do not belong to any cluster. Let the edges corresponding to these vertices be referred to as inter-cluster edges. If an

---

[8] $\lceil x \rceil$ is the smallest integer greater than or equal to x.

edge is not an inter-cluster edge, then it is between vertices in a single cluster. Such an edge is termed an intra-cluster edge. In these cluster definitions it is reasonable to expect the clusters to remain unaltered if there is a decrease in the number of inter-cluster edges. We specify this as a property that is required of any reasonable cluster definition.

Definition 5.2.5: A cluster definition is cohesively consistent if the deletion of the edges between vertices that are not in the same cluster (inter-cluster edges) leaves all the clusters unchanged.

In the rest of the chapter, it is assumed that a cluster definition is either adjacency oriented or cohesively consistent and that in either case the definition generates unfragmented clusters.

5.2.2 The Main Result Concerning Adjacency Oriented Clustering Methods

In order to show that clusters defined as the CC's are the most stable of all adjacency oriented definitions, we specify some further notation. The graphs G and G* represent, as mentioned, the graphs before and after certain input changes, and G** denotes the restored graph that has the same clusters as G. Let G have e edges, s components with n_j vertices in the j^{th} connected component, g_j, for $1 \leq$

$j \leq s$, and let $n_j \geq n_{j+1}$, for $1 \leq j \leq (s-1)$. The set of graphs that have these properties is denoted by $\mathcal{G}(s, e, n_1, n_2, \ldots n_s)$. Finally, we denote the subgraphs consisting of the vertices of g_j in G* and G** respectively by g_j^* and g_j^{**}. Note that g_j^* need not be connected.

The proof of the proposition concerning the relative stability of the CC definition consists, essentially, of showing that G and G** can have the same clusters according to any adjacency oriented definition only if, corresponding to each component in G, the graph G** has a component consisting of exactly the same vertices. Thus, as is required for the CC definition, restoring of clusters according to any adjacency oriented definition requires that all edges added between the components of G be removed and that enough intra-cluster edges be added to g_j^* so that g_j^{**} becomes connected. Lemma 5.2.2 gives the amount of work required to restore any g_j^* that might be disconnected, to g_j^{**}. Lemma 5.2.3 is a technical result used in proving proposition 5.2.4.

Lemma 5.2.2: Let G have s components. Suppose that the perturbations in the data are such that no inter-component edges are added to G and that g_j in G, $1 \leq j \leq s$, are split into m_j components in G*. Then $W(CC) = \sum_{j=1}^{s} (m_j - 1)$.

Proof: It is easy to show, by induction on the number of components, that the minimum number of edges needed to combine m_j components into one is $(m_j - 1)$. #

Lemma 5.2.3: Let H = (V,E) be a connected graph and let H' = (V,E') be another graph on the same vertex set. If v is connected to w in H', for every v, w $\in$ V such that v is adjacent to w in H, then H' is connected.

Proof: Let a, b be any two vertices in H'. Since H is connected, there exists a path from a to b. Suppose it is a, v_1, v_2, ..., v_j, b. Then, by hypothesis, in the graph H', a is connected to v_1, v_i is connected to v_{i+1} for $1 \leq i \leq j-1$ and v_j is connected to b. Thus, a is connected to b in H'. Since the above result is true for any a, b $\in$ V, H' is connected. #

Proposition 5.2.4: Given a cluster defining method D, W(CC) $\leq$ W(D) for every initial graph G and for every set of changes to G if and only if the cluster definition D is adjacency oriented.

Proof: ==>: If D is not adjacency oriented, then there exists a graph and a set of changes to it such that W(D) < W(CC). Specifically, consider a graph G = (V,E) such that

i) the edge (u,v) $\in$ E is a bridge, and

ii) the vertices u and v do not belong to any cluster according to D.

If the only change to G is the "deletion of the edge (u,v)", then it is clear that W(CC) = 1. Since D is cohesively consistent, W(D) = $\emptyset$.

<==: The changes causing G to be modified to G* can

be considered to be one of the following three types.

Case 1: The set of changes specified by

C_1 = {changes in the adjacency relationship between v, w $\in$ V | v is connected to w in both g_j and g_j*}.

The changes of this type have no effect on W(CC). But, the effect on W(D) is non-negative. Thus, W(D) $\geq$ W(CC) (due to C_1).

Case 2: The addition of inter-cluster edges, given by

C_2 = {addition of edges e $\in \overline{E}$ to G | if ϕ(e) = v & w $\in$ V, then v is not connected to w in G}.

For restoring clusters according to the CC definition, each such edge must be deleted. But, using the definition D, v and w cannot be assigned to the same cluster in G since D is unfragmented. On the other hand, there will be at least one cluster in G* containing v and w since D is adjacency oriented. Therefore, each such edge must be deleted for proper restoration of the clusters according to definition D.

Case 3: The intra-cluster edge deletions that break up some components, given by

C_3 = {deletion of edges e in G | if ϕ(e) = v & w $\in$ V, then v is not connected to w in the g_j* containing v and w}.

The above set can be partitioned into s subsets as follows:

C_{3j} = {deletion of edges e in g_j | if $\phi(e) = v$ & $w \in V$, then v is not connected to w in g_j^*}.

That is, the removal of the set C_{3j} causes g_j to split into a number, say $(m_j + 1)$, of components. By lemma 3.2, the restoration of the cluster g_j with respect to the CC definition requires m_j edge additions. That is, W(CC) increases by m_j, $1 \leq j \leq s$.

Consider the restoration process with respect to the definition D. Since D is adjacency oriented, for any pair of adjacent vertices (v,w) in g_j there exists a cluster containing v and w. In G**, there is a corresponding cluster containing v and w. Since clusters are required to be unfragmented, v must be connected to w in G**. Since this is true for every pair of adjacent vertices in g_j, by lemma 5.2.3, the set of vertices in g_j must be connected in G**. Since g_j^* has $m_j + 1$ components, a minimum of m_j edges must be added to g_j^* to make these vertices connected (lemma 5.2.2). Thus W(D) due to C_{3j} is $\geq$ W(CC), $1 \leq j \leq s$.

Thus, putting all three cases together, we have $W(D) \geq W(CC)$. #

Note that cases 2 and 3 considered above imply that for

an adjacency oriented definition, D, there is a component wise correspondence between G and G**. That is, G** $\in$ $\mathcal{G}(s,e_1,n_1,n_2, \ldots, n_s)$, where e_1 is the number of edges in G**.

5.2.3 Stability Ordering in Adjacency Oriented Families of Clustering Methods

We next consider the different families of adjacency oriented cluster definitions described earlier. As these definitions are adjacency oriented and unfragmented, we have

Corollary 5.2.5: For every G and every set of modifications to G,

$W(MCS) \geq W(D) \geq W(CC)$, where D is any adjacency oriented clustering method.

Proof: Obvious, given propositions 5.2.1 and 5.2.4, and since any pair of adjacent vertices must be in at least one cluster according to D. #

A related problem is that of assessing the stability of the different members of a particular sequence of clustering methods relative to each other. In other words, it is of interest to know, for instance, whether or not $W(B_k) \leq W(B_{k+1})$. The property in question is defined below.

Definition 5.2.6: Suppose A_t denotes a sequence of adjacency oriented graph theoretic cluster defining methods with the

property that the clusters generated by these schemes are nested in the sense that any cluster defined by A_{t_2} must be included in some cluster defined by A_{t_1} for any $t_1 < t_2$. Furthermore, let each member A_t be MCS centered and unfragmented. Such a sequence A_t is stability ordered if $W(A_{t_1}) \leq W(A_{t_2})$, for $t_1 < t_2$.

It is easy to show by counter example that the sequence D_t, defined earlier, is not stability ordered. Consider the graphs in **Figure 4.** Let G and G* represent the original and the modified graphs respectively. For D_t with t=0.8 the clusters in both G and G* are {(1,2),(2,3,4,5)}. Thus, no restoration is needed in this case. However, if t=0.65 then the clusters defined by D_t on G are different from that on G* implying that some restoration must be performed. Clearly, for this example, $W(D_t,t=0.65) > W(D_t,t=0.8)$.

It is now shown that the methods B_k are stability ordered. The proof requires a precise statement of the algorithm that generates B_k clusters. Let G(V,E), |V|=p, be the given graph and, for a given k, let $G_k=(V,E_k)$ and $G_{k+1}=(V,E_{k+1})$ be graphs such that their MCS'S are exactly the clusters generated, respectively, by B_k and B_{k+1} for an input graph G. Note that, by proposition 5.2.1, G_k and G_{k+1} are unique. G_k whose MCS's are the clusters of B_k is recursively defined below:

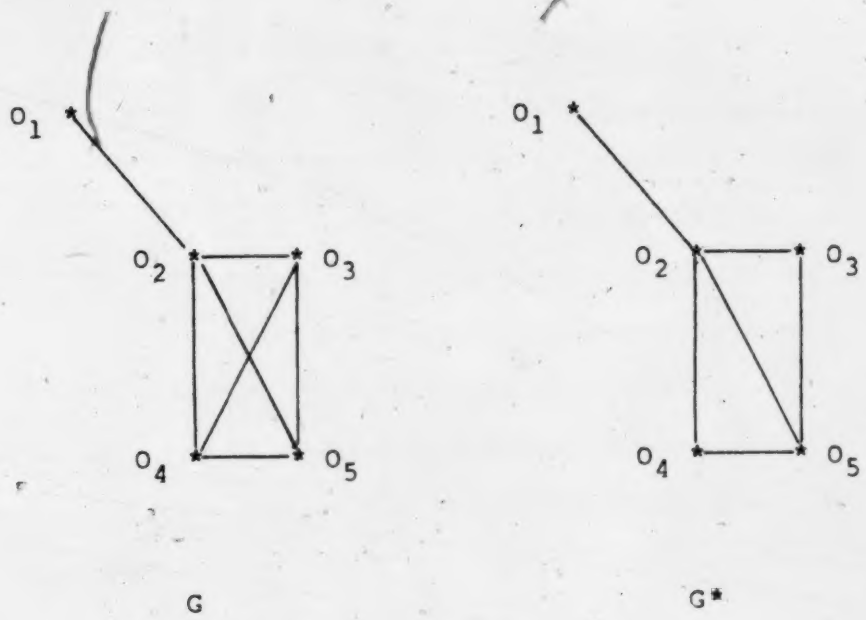


FIGURE 4. An initial and a perturbed graph to illustrate D_t is not stability ordered.

Procedure CLUSTER-B_k (G)

Begin

If $k \geq p-1$ then return G

else Begin

CLUSTER-B_{k+1} (G) (The graph returned is G_{k+1}. Let $H = (V, E_H)$ be G_{k+1}.);
While (there exists a pair of MCS's M_1, M_2 in H such that M_1 and M_2 have k or more vertices in common)
Do insert edges necessary to make $M_1 \cup M_2$ into an MCS (that is, $E_H = E_H \cup$ { the inserted edges});
Return the resulting graph $H = G_k$;

end

end.

Let $G^* = (V, E^*)$ denote the graph obtained after some perturbations have occurred in the original graph. Let $W(B_k)$ and $W(B_{k+1})$ be respectively the minimum amount of work required to change G^* to some G_k^{**} and G_{k+1}^{**} such that CLUSTER-B_k (G_k^{**}) returns G_k and CLUSTER-B_{k+1} (G_{k+1}^{**}) returns G_{k+1}. We establish that the fine clustering methods, B_k, are stability ordered by showing that the set of edge corrections made to G^* to form G_{k+1}^{**} are sufficient also to obtain G_k by using the algorithm CLUSTER-B_k.

Proposition 5.2.6: The sequence B_k of fine clustering methods is stability ordered.

Proof: By definition of the procedure CLUSTER-B_k,

$$\text{Cluster-}B_k(G^{**}_{k+1}) = \text{CLUSTER-}B_k\ (\text{CLUSTER-}B_{k+1}\ (G^{**}_{k+1}))$$

$$=\text{CLUSTER-}B_k(G_{k+1}) = G_k. \qquad \#$$

The algorithms for methods B^c_k, C_u and the method proposed by Gotlieb and Kumar for the merging of MCS's are defined, essentially, in the same way. Therefore, the proof is easily generalized to also show that these sequences are stability ordered.

5.3 The Maximum and the Minimum Amount of Work for Clusters Defined as the CC's

In the last section it was shown that clusters defined as the CC's are the most stable, in terms of the measure of stability suggested, of all definitions that are adjacency oriented and that a number of families of cluster definitions possess this property. Therefore, it is of interest to study the clusters defined as the CC's in more detail.

We assume that the original graph is a member of $\mathcal{G}(s,e,n_1,n_2,\ldots.n_s)$. An upper bound and a lower bound for the amount of work required to restore clusters defined as the CC's are then obtained in terms of the parameters s, e, $n_1,n_2,\ldots.n_s$, a, the number of edges added to the original graph to obtain the modified graph (which is $|E^*-E|$) and d, the number of edge deletions from G (given by $|E-E^*|$).

5.3.1 The Lower Bound

When edges are deleted from a connected component, the component may be broken up into smaller components. To achieve the most stable case, however, the edges that are not essential to maintain the connectivity of a component should be chosen for deletion.

Let us consider the effect of edge addtions next. The addition of edges can occur within or between components. Whenever an inter-cluster edge is added, one operation has to be performed to get rid of the edge. However, when intra-cluster edges are added, there is no disturbance in the clusters obtained. Thus, in obtaining the lower bound, if the number of additions is small enough ($a \leq \sum_{j=1}^{s} C(n_j,2) - e$) to occur within components, then the edges for addition will be chosen in such a way that the number of inter-cluster additions is zero. Consequently, we have the following proposition.

Proposition 5.3.1: For any a, d and $G \in \mathcal{G}(s, e, n_1, \ldots n_s)$

$$W(CC) \geq \begin{cases} a' & \text{if } d \leq z \\ a' + (d - z) \dot{-} (a - a') & \text{if } d > z \end{cases}^{9}$$

---

[9] $x \dot{-} y = x - y$ if $x > y$ and zero otherwise.

where $z = e - \sum_{j=1}^{s}(n_j - 1)$ and $a' = a \div \left(\sum_{j=1}^{s} C(n_j,2) - e \right)$

Proof: a' is the number of inter-cluster edge additions. Each such edge must be deleted. Thus, W(CC) $\geq$ a' in both cases ($d \leq z$ and $d > z$). Let e_j denote the number of edges in g_j. Since $(n_j - 1)$ edges are sufficient to maintain g_j connected, it is possible to delete $z_j = e_j - (n_j - 1)$ edges and still guarantee that g_j remains connected. Summing over j from 1 to s, it is clear that W(CC) due to edge deletions equals zero when the number of deletions $d \leq z$. After z deletions, we have s minimally connected components. Therefore, each deletion beyond the z deletions causes 1 unit of work. However, these deletions may be offset by the intra-cluster additions which number (a-a').

#

When $d \leq z$, it is easy to see that the deletions can occur in such a way that the remaining edges are enough to ensure that each component stays connected. For $d > z$, however, it is not clear if, for any integer $x \leq (d - z)$, a set of x edge additions can be specified such that each such addition will always offset the effect of a deletion beyond z. It turns out that this offsetting can be brought about by ensuring that the added edges do not form a cycle. Of course, if the number of edges added to g_j is greater than $(n_j - 1)$ then one or more cycles must occur. In this case,

however, the number of additions involved is sufficient to make g_j connected and can therefore offset all the deletions in g_j. In the following lemma, it is shown that if the edges to be added, $|\bar{E}|$, do not form a cycle then we can ensure that the resulting graph is connected for any number of deletions, $|T|$, up to the number of edges added. The tightness of the bound is then proved by identifying a graph for which the lower bound is attained.

Lemma 5.3.2: Let $\bar{H} = (V, \bar{E})$ and $H'' = (V, E'')$ be respectively the complement graph and a minimally connected subgraph of $H = (V,E)$. If $\bar{H}$ has no cycles, then there exists a set of edges T in H" such that, whenever $|T| \leq |\bar{E}|$, $H^{**} = (V, \bar{E} \cup E'' - T)$ is connected.

Proof: Let $t = |T|$. Consider a set of t edges in $\bar{H}$. Add any one of the edges to H". Exactly one cycle is created in H". This cycle must contain an edge originally in H". Clearly, this edge can be deleted (i.e. assigned to T) without affecting the connectivity of H". This process is then repeated for the t edges in $\bar{E}$. Each time a cycle is created in H", by the hypothesis that $\bar{H}$ has no cycles, the cycle must contain at least one edge originally in H", and this edge can be deleted while maintaining the connectivity of H**. #

The above result shows that the effect of edge deletions up to $|\bar{E}|$ in number can all be offset by $|\bar{E}|$ edge

additions.

Proposition 5.3.3: The lower bound of proposition 5.3.1 is the tightest.

Proof: Case (i) $d \leq z = e - n + s$; obvious.

Case (ii) $d > z$.

Consider g_j, the j^{th} component of G. Let a_j and d_j be respectively the number of additions and deletions to this subgraph. Clearly, $\sum_{j=1}^{s} a_j = a - a'$. Let $\overline{g}_j$ be the complement graph of g_j, $1 \leq j \leq s$, and $\overline{G}$ the union of all the $\overline{g}_j$'s.

Assume that G is chosen in such a way that if there are sufficient number of edges in $\overline{G}$ then each $\overline{g}_j$ is connected; otherwise each $\overline{g}_j$ has no cycles. The first z deletions are made from G such that the resulting graph, G", is minimally connected. That is, as indicated in proposition 5.3.1, z_j deletions transform g_j to a minimally connected component g_j". We now consider the effect of a_j additions and $(d_j - z_j)$ deletions on g_j". If the number of edges in $\overline{g}_j$ is less than or equal to (n_j-1) then let g_j" and $\overline{g}_j$ correspond respectively to H" and $\overline{H}$ of lemma 5.3.2. Otherwise, let g_j" and a minimally connected subgraph of $\overline{g}_j$ correspond to H" and $\overline{H}$ of the lemma.

If $a_j \geq (n_j - 1)$, then these edges are

sufficient to make the j^{th} component connected. Thus, no work is needed irrespective of d_j. If $a_j < (n_j - 1)$, then either $a_j \geq (d_j - z_j)$ or $a_j < (d_j - z_j)$. In the former case, by lemma 5.3.2, the graph obtained after the additions and deletions remains connected. Thus, no work is involved. In the latter case, the graph can remain connected up to $(a_j + z_j)$ deletions and a_j additions (by lemma 5.3.2). Each deletion after that increases the number of components in the subgraph by one. Thus, the total amount of work is $\sum_{j=1}^{s} (d_j - z_j - a_j)$, which is $d - z - (a - a')$. #

5.3.2 The Upper Bound

In order to obtain the upper bound the additions must occur between components whereas deletions must break up the components in the original graph. For the sake of presentation, the number of edge additions is initially assumed to be zero. The general situation will be considered later in this section.

If H denotes a connected component of a graph from which certain edges are deleted, then in the perturbed graph either the vertices of H are still connected or they split into a number of components. By lemma 5.2.2, it is clear that the larger the number of components in the perturbed graph, the more is the work required to do the restoration. The above relationship suggests that one of the critical

steps in deriving the upper bound is to determine the maximum number of components that a graph with certain specified characteristics can have. Given n vertices and e edges, suppose we wish to construct a graph with the maximum number of components. An intuitive approach to the problem is to concentrate the edges between as few vertices as possible. That is, a single component which is just large enough to contain all the edges is formed and the remaining vertices are left isolated (this problem is, in fact, the dual of maximizing the number of edges in fixed number of components and vertices).

In the problem considered above no restriction has been placed on how the edges should be assigned to the vertices. But, in our context, the original graph G, contains s components. After d edge deletions, the j^{th} component g_j becomes g_j^*. Since g_j^* may be disconnected, we refer to g_j^* as a <u>part</u> of G*. Our aim is to find a graph having the maximum number of connected components. This graph must have s parts (corresponding to the s components of G) such that the j^{th} part consists of n_j vertices, $1 \leq j \leq s$, a total of (e - d) edges in the s components and $n_j \geq n_{j+1}$, for $1 \leq j \leq (s-1)$. Let the set of graphs with the above characteristics be denoted by $\hat{g}(s, e-d, n_1, n_2, \ldots, n_s)$. A natural generalization of the scheme referred to earlier is to use the (e - d) edges to first fill the part with the largest number of vertices, then the next largest, and so on, until all the edges are exhausted. This basic idea is formalized,

and a proof that it leads to the maximum number of components is presented.

Definition 5.3.1: A graph H = (V,E) where |V| = n and |E| = e is partially complete if $C(n-1,2) + 1 \leq e \leq C(n,2)$.

Definition 5.3.2: A part consisting of m vertices, of a graph H = (V,E) $\in \hat{\mathcal{G}}$, is k-compact if it contains a partially complete subgraph of (k + 1) vertices and m - (k + 1) isolated vertices.

Definition 5.3.3: A graph G = (V,E) $\in \hat{\mathcal{G}}(s, e, n_1, n_2, \ldots, n_s)$ is (i,k)-compact if the first (i - 1) parts, $1 \leq i \leq s$, are complete subgraphs, part i is k-compact, $1 \leq k \leq n_i$, and parts i+1, ..., s are isolated vertices.

Lemma 5.3.4: Let $\tilde{G}$ = (V,E) $\in \hat{\mathcal{G}}(s, e, n_1, n_2, \ldots, n_s)$ be an (i,k)-compact graph. Then, e satisfies

$$C(k,2) + 1 \leq e - \sum_{j=1}^{i-1} C(n_j,2) \leq C(k+1,2) \qquad (5.3.1)$$

and $\tilde{G}$ has at least as many components as any graph in $\hat{\mathcal{G}}(s, e, n_1, n_2, \ldots, n_s)$.

Proof: It is clear that the inequality (5.3.1) is satisfied, since a complete subgraph in x vertices has C(x,2) edges. By simple computation, $\tilde{G}$ has $i - 1 - k + \sum_{j=i}^{s} n_j$ components.

To show that this is the maximum, suppose that a graph different from $\tilde{G}$ from the set $\hat{\mathcal{G}}(s, e, n_1,$

$n_2, \ldots, n_s$) has a greater number of components. Let the j^{th} part in this graph have $n_j - x_j$ components, $0 \leq x_j \leq (n_j - 1)$. Then, by hypothesis,

$$\sum_{j=1}^{s} (n_j - x_j) > i - 1 - k + \sum_{j=i}^{s} n_j.$$

This implies that

$$\sum_{j=1}^{s} x_j \leq \sum_{j=1}^{i-1} (n_j - 1) + (k-1) \qquad (5.3.2).$$

By theorem 2.3 in (Deo, 1974), a graph of $(n_j - x_j)$ components and n_j vertices has no more than $C(x_j,2)$ edges. Thus, there are at most $C(x_j,2)$ edges in the j^{th} part. Consequently, the number of edges in the graph,

$$e \leq \sum_{j=1}^{s} C(x_j,2)$$

$$= 1/2 \left(\sum_{j=1}^{s} x_j^2 - \sum_{j=1}^{s} x_j \right).$$

Given inequality (5.3.2) and assuming without loss of generality that $n_j > n_{j+1}$, $1 \leq j \leq (s-1)$, by corollary B.2 (appendix B) we have

$$e \leq 1/2 \left\{ \sum_{j=1}^{i-1} (n_j - 1)^2 + (k - 1)^2 \right.$$

$$\left. - \sum_{j=1}^{i-1} (n_j - 1) - (k - 1) \right\}$$

$$= \sum_{j=1}^{i-1} C(n_j,2) + C(k,2).$$

This contradicts the inequality (5.3.1). #

We now specify a graph in the set $\mathcal{G}(s, e, n_1, n_2, \ldots, n_s)$ such that after d edge deletions are made the resulting graph requires the maximum amount of work for restoration. It turns out that the graph in $\mathcal{G}(s,e,n_1,n_2,\ldots,n_s)$ that attains the upper bound is the one in which as many of the larger parts as possible are maximally connected and the remaining parts are minimally connected. The edge deletions are assumed to occur in the minimally connected parts whenever possible. Each such deletion causes one unit of work and, clearly, the work involved in this case is the maximum possible. When the edges in the minimally connected parts are exhausted, the deletions will occur in such a way that the resulting graph is an (i,k)-compact graph. This process ensures that as many connected components as possible will be obtained. The following definitions and propositions make these ideas precise.

Definition 5.3.4: Let g = (V', E'), where |V'| = m and |E'| = e, be a component of a graph. Suppose for some k, $2 \leq k \leq (m-1)$, $C(k,2) + 1 \leq e - (m - k - 1) \leq C(k+1,2)$. Then, g is k-packed if it contains a partially complete subgraph of (k + 1) vertices, a minimally connected subgraph in the remaining (m-k-1) vertices and exactly one edge between the

two subgraphs.

Definition 5.3.5: A graph $G = (V,E) \in \mathcal{G}(s, e, n_1, n_2, \ldots, n_s)$ is (i,k)-packed if the first $(i-1)$ components, $1 \leq i \leq s$, are complete subgraphs, component i is k-packed, $2 \leq k < n_i$, and components $i+1, \ldots, s$ are minimally connected.

Note that the number of edges, e, in an (i,k)-packed graph satisfies the following inequality:

$$\sum_{j=1}^{i-1} C(n_j,2) + C(k,2) + 1 \leq e-p \leq \sum_{j=1}^{i-1} C(n_j,2) + C(k+1,2),$$

where p is the number of edges in the minimally connected sections of $\hat{G}$ (that is, $p = \sum_{j=i}^{s}(n_j-1)-k$).

Proposition 5.3.5: Let $\hat{G} = (V,E) \in \mathcal{G}(s, e, n_1, \ldots, n_s)$ be an (i, k)-packed graph. Then, for any given number of deletions, d, the maximum amount of work required for the restoration is attained for $\hat{G}$.

Proof: Case (i) $1 \leq d \leq p$.

That is, the number of edge deletions is not more than the number of edges in the minimally connected parts of $\hat{G}$. Consequently, each deletion in this range increases the number of components by 1. Clearly, this is the maximum possible.

Case (ii) $p < d \leq e$.

It is clear that the deletion of the d edges can be made from $\hat{G}$ such that the modified graph is an (i,k)-compact graph in $\hat{\mathcal{G}}(s, e-d, n_1, n_2, \ldots, n_s)$. By lemma 5.3.4, this graph has the maximum number of components. By lemma 5.2.2, the number of edge additions required for restoration is the maximum. #

We now present the expressions for the upper bound for the different possible cases of d. Initially, each deletion causes 1 unit of work. Then, the $(k+1)^{st}$ vertex of the i^{th} component is isolated. The isolation of this vertex may involve a number of deletions. The other cases consider situations in which some edges are to be deleted from complete subgraphs. For these cases, the amount of work is expressed as a function of the number of edges deleted from a complete subgraph, and the number of vertices in the complete subgraph.

Proposition 5.3.6: Let $G \in \mathcal{G}(s, e, n_1, n_2, \ldots, n_s)$ be such that for some i, $1 \leq i \leq s$, and for some k, $2 \leq k < n_i$, e satisfies

$$\sum_{j=1}^{i-1} C(n_j,2) + C(k,2) + 1 \leq e - p \leq \sum_{j=1}^{i-1} C(n_j,2) + C(k+1,2).$$

If the perturbation involves d edge deletions, then

$$W(CC) \leq \begin{cases} d \text{ if } d \leq p \text{ (minimally connected portion)} \\ p \text{ if } p < d < q \text{ (edges between the } (k+1)^{st} \text{ vertex and the complete subgraph of } k \text{ vertices in } i^{th} \text{ part)} \\ p + 1 \text{ if } d = q \text{ (the } (k+1)^{st} \text{ vertex above is isolated)} \\ p + 1 + f(k, d - q) \text{ if } q < d \leq b_{i-1} \text{ (edges in the complete subgraph of size } k \text{ in the } i^{th} \text{ part)} \\ \sum_{j=t+1}^{s} (n_j - 1) + f(n_t, d - b_t) \text{ if } b_t < d \leq b_{t-1} \text{ (edges in the } t^{th} \text{ part for some } t) \end{cases}$$

where for t, $1 \leq t \leq i-1$,

$$b_t = e - \sum_{j=1}^{t} C(n_j, 2) \text{ (where } b_0 = e),$$

$$q = b_{i-1} - C(k,2) \text{ (number of edges not in the maximally connected sections of } \hat{G} \text{)}$$

and

$f(x,y)$ is the minimum amount of work to restore a cluster, which is originally a complete subgraph of size x, after y deletions occur. $f(x,y)$ is shown in lemma B.3 (appendix B)

to be $\leq 1 + 4y / (2x-1)$, which says that for each deletion the amount of work for restoration is, roughly, $(2/x)$.

Proof: Obvious. #

Suppose the perturbation involves both edge additions and edge deletions. The upper bound for the amount of work, due to additions, will be achieved if the perturbations resulting in edge additions occur between components since every edge so added must finally be removed. Thus, if the number of edge additions is not so large that all additions can occur between components, then the bounds of proposition 5.3.6 will be increased by the quantity a', the number of inter-cluster additions. When the number of intra-cluster edge additions, (a - a'), is not zero it is easy to see that the maximum work will be attained for an (i',k')-packed graph of (e + a - a') edges. Consequently, the values of p, q, and b_j, $1 \leq j \leq i-1$, of proposition 5.3.6 will have to be appropriately modified.

5.3.3 A Comparison of the Bounds

In this section the results concerning the lower and the upper bounds are summarized. Given a graph $G \in \mathcal{G}(s,e,n_1,\ldots,n_s)$, let the changes involve d deletions and a additions. It was mentioned earlier that to achieve the lower bound, whenever possible, the additions must be within components; for the upper bound, however, the additions would be between components. For convenience, assume that

the number of additions of these two types are held constant independently of each other. That is, let a' be the number of additions between components and (a - a') the number of additions within.

Figure 5 characterizes the bounds of the CC definition relative to the actual amount of work needed for the MCS definition. The line for W(MCS) shows that each deletion increases the amount of work by 1. The upper bound for W(CC) overlaps with the line for W(MCS) up to a certain point, and then increases at a decreasing rate. The values of p', q' and bj', $1 \leq j \leq i-1$, are obtained using the steps in proposition 5.3.6, but in reference to an (i',k')-packed graph $\in \mathcal{G}(s, e+a-a', n_1, \ldots n_s)$. In other words, when (a - a') intra-cluster edges are added to an (i,k) packed graph in e edges (in such a way that the desired structure is retained) the values of i and k may be modified to i' and k' respectively. The lower bound for W(CC) initially remains at a' and, after (d - z - (a - a')) deletions occur, increases at the same rate as W(MCS) provided $(a - a') \leq \sum_{j=1}^{s} (n_j - 1)$. If $(a - a') > \sum_{j=1}^{s} (n_j - 1)$ then, as shown in proposition 5.3.3, W(CC) remains at a' irrespective of d.

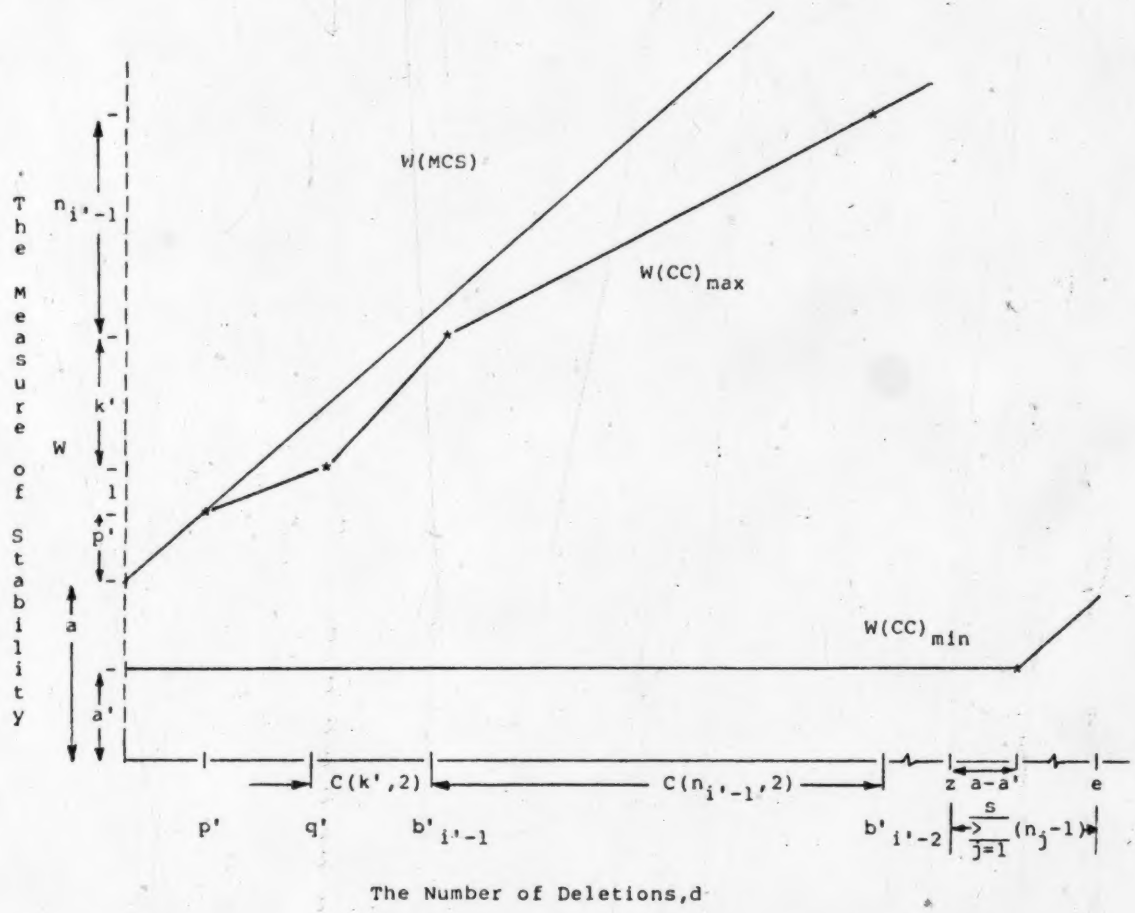


FIGURE 3. Stability of graphs in $\mathcal{G}(s, e, n_1, n_2, \ldots n_s)$ for d deletions, a' inter- cluster additions and (a-a') intra-cluster additions. p', (e-q') and b_j', $1 \leq j \leq i'-1$, are respectively the number of edges in the minimally connected sections, the maximally connected sections and the first (i'-1) components of an (i',k')-packed graph.

Chapter 6

SUMMARY OF FINDINGS AND PROPOSALS FOR FURTHER RESEARCH

A review of techniques for the classification of terms indicates, for the most part, that the utility of currently available fully automatic methods is, at best, marginal. In this regard, Salton (1972a) states

> "For the present time, a combination of manual and automatic thesaurus methods therefore appears most promising for practical applications, involving the following steps:
> a. automatic common word recognition;
> b. manual term classification;
> c. automatic refining of the manually produced classes."

Typical examples of such thesaurus refinement are the methods of Gotlieb and Kumar (1968) and Dattola and Murray (1967). A vast majority of these semi-automatic and fully automatic methods measure the similarity between terms based on the hypothesis that the more often two terms tend to cooccur, the more likely they are to be substitutable for each other.

In this thesis, an automatic method for determining the relationships between terms is proposed. The method uses relevance judgments obtained in relation to a sample of user queries. Currently available methods for the classification of terms only identify similar terms in the sense of terms being synonymous or being substitutable for each other. In contrast, the method proposed in this work can also identify terms that are dissimilar in meaning or in the context that they provide to documents and queries. The term

relationships obtained are incorporated into the retrieval process by using a generalized similarity function. More specifically, whereas the basic similarity function determines the closeness between a document and a request on the basis of only the matches in the index terms, the generalized function also takes the relationships between terms into account.

Two strategies that differ in the scope of term relationships that they establish are evaluated. The relationships between terms derived from one of these strategies is considered global since the set of term relationships obtained are applicable to the collection of documents and queries as a whole. On the other hand, the other strategy obtains localized term relationships which are applicable only to a relatively more homogeneous subcollection of the documents and the queries. Irrespective of the scope of the relationships between terms, the retrieval performance when these relationships are used is found to be better than when they are not. Furthermore, it is found that the localization of the term relationships results in more effective retrieval than that of using global relationships. The former strategy is also more attractive since it requires a smaller amount of storage.

The effect of document frequencies of terms on the significance of their relationships to other terms is investigated. Intuitively, since a term occurring in many

documents has a fairly broad scope, the relationships it has to other terms is likely to be vague or imprecise. On the contrary, terms that occur in a small number of documents, because of their narrower scope, may exhibit less variability in their meaning. Thus, the relationships between terms that occur in just a few documents may be expected to be more significant for retrieval purposes. Experimental results obtained confirm this intuitive assessment.

One of the difficulties in performing experimental investigations of document retrieval processes stems from the fact that these processes are usually determined by numerous parameters. Since there exist a large number of reasonable combinations of values for the parameters, a very large number of experiments may have to be run before proper values can be selected. Thus, the experimental analysis often becomes very tedious. In the current problem, the generalized similarity function used to measure the closeness between a document and a request consists of three components; a term match component reflecting the extent to which index terms chosen by the indexer match, a positive component which measures the extent to which terms in the document are similar to those in the query and a negative component measuring the dissimilarity between the terms in the two items. Clearly, it is important to determine, by experimentation or otherwise, the relative importance of these components in achieving improved retrieval

performance. It is shown that a multivariate statistical method known as discriminant analysis can be used to assess the discriminating power of the components. This approach reduces the number of parameters that have to be analysed experimentally. The analysis shows that the term match component which is based on indexer provided information is at least as important as the two term relationship components combined. Furthermore, the positive component is found to have more discriminating ability than the negative component.

The computing time for the proposed method is roughly proportional to the number of document-request pairs processed. This aspect renders the approach viable in a practical environment.

In comparing the proposed method to the earlier methods, based on the cooccurrences of terms, it is easily seen that the information obtained from the cooccurence statistics is quite distinct from the relevance information that our method uses. It may, therefore, be worthwhile to consider methods by which a retrieval system can benefit from both kinds of information. The proposed method also differs in the sense that positive and negative relationships between terms are determined. Thus, the idea of constructing classes of terms is not appropriate. This aspect creates difficulties in interpreting the meaning of independence of scale in our context. Furthermore, since the

proposed method is order dependent, the commonly used approaches to the assessment of stability are not applicable to our situation. These problems may be investigated in a future study.

The literature survey of the various methods of classification also reveals the predominance of graph theoretic clustering schemes in document retrieval applications. Many studies have evaluated such methods from the point of view of effectiveness and efficiency. In the current study the stability of a number of commonly used graph theoretic clustering methods is analysed.

A measure of stability that determines the disturbance in a set of clusters as the minimum number of operations required to restore the set of modified clusters to the original ones is adopted. A number of families of graph theoretic cluster defining schemes, referred to as adjacency oriented, are shown to include two well known methods of defining clusters: connected components and maximal complete subgraphs. The connected component method is shown to be more stable than any adjacency oriented clustering method irrespective of the initial graph and the perturbations that take place. The maximal complete subgraphs are found to represent the worst possible case in terms of this measure of stability. Furthermore, it is shown that for certain families of adjacency oriented clustering methods the various members in the family are ordered with respect to

the measure of stability. A lower and an upper bound for the measure of stability, when clusters are defined as the connected components, are derived. These bounds are the tightest possible.

Analysing the stability of the clustering schemes that are not adjacency oriented in terms of the the measure of stability adopted in this work is a very promising area for further research.

REFERENCES

Abraham, C. T. (1965a). Techniques for thesaurus organization and evaluation. In: Kochen (Ed.), Some Problems in Information Science, pp. 131-150, The Scarecrow Press, Newyork, N.Y.

Abraham, C. T. (1965b). Graph theoretic techniques for the organization of linked data. In: Kochen (Ed.), Some Problems in Information Science, pp. 229-264, The Scarecrow Press, Newyork, N.Y.

Anderberg, M. R. (1973). Cluster Analysis for Applications. Academic Press Inc., Newyork, N. Y.

Anderson, T. W. (1958). Introduction to Multivariate Statistical Analysis. Wiley, Newyork, N.Y.

Auguston, J. C. and Minker, J. (1970a). An analysis of some graph theoretical cluster techniques. JACM., 17, 571-588.

Auguston, J. C. and Minker, J. (1970b). Deriving term relations for a corpus by graph theoretical clusters. J. of the American Society for Information Science, 21, 101-111.

Ball, G. H. (1965). Data analysis in social sciences: what about details. Proceeedings AFIPS. FJCC, Macmillan, Newyork, N. Y., 533-559.

Bonner, R. E. (1964). On some clustering techniques. IBM J. of Research and Development, 8, 22-32.

Cain, A. J. (1958). Logic and memory in Linnaeus's system of taxonomy. Proceedings Linn. Soc. Lond., 169th session, 144-163.

Chan, F. K. (1973). Document Classification through Use of Fuzzy Relations and Determination of Significant Features. M. Sc. thesis, Department of Computing Science, The University of Alberta.

Cleverdon, C. W. and Keen, E. M. (1966). Factors Determining the Performance of Indexing Systems. Aslib-Cranfield Research Project Report, Vol. 1 and 2, Cranfield, England.

Cole, A. J. and Wishart, D. (1970). An improved algorithm for the Jardine-Sibson method of generating overlapping clusters. Computer J., 13,

156-163.

Cooley, W. W. and Lohnes, P. R. (1971). Multivariate Data Analysis , John Wiley & Sons, Inc., Newyork, N.Y., 243 - 261.

Cormack, R. M. (1971). A review of classification. J. Royal Statistical Society - Series A, 134, 321-367.

Corneil, D. G. and Woodward, M. E. (1978). A comparison and evaluation of graph theoretical clustering techniques. Infor, 16, 74-89.

Dattola, R. T. and Murray, D. M. (1967). An experiment in automatic thesaurus construction. In: Information Storage and Retrieval, Scientific Report No. ISR-13, Chpt. VIII, Cornell University, Ithaca, N. Y.

Day, W. H. E. (1977). Validity of clusters formed by graph-theoretic cluster methods. Mathematical Biosciences, 36, 299-317.

Deo, N. (1974). Graph Theory with Applications to Engineering and Computer Science. Prentice-Hall, Englewood Cliffs, N.J., 22-23.

Diogenes, L. (1925). Lives of Eminent Philosophers (Hicks, Transl.). Loeb Classical Library, 2 Vols., W. Heinemann, London.

Doyle, L. B. (1961). Semantic road maps for literature searchers. JACM., 8, 553-578.

Doyle, L. B. (1966). Breaking the Cost Barrier in Automatic Classification. Report No. SP-2516, System Development Corp., Santa Monica, Calif.

Gotlieb, C. C. and Kumar, S. (1968). Semantic clustering of index terms. JACM. 15, 493 - 513.

Guiliano, V.E. and Jones, P. E. (1963). Linear associative information retrieval. In:Howerton and Weeks (Eds.), Vistas in Information Handling , Vol. 1, Ch. 2, Spartan Books, Washington, D. C.

Hubert, L. J. (1974). Some applications of graph theory to clustering. Psychometrika, 39, 283-309.

Jackson, D. M. (1969a). An error analysis for functions of qualitative attributes with applications to information retrieval. In: Tou (Ed.), Software Engineering - COINS III, Vol. 2,

Proceedings of the Third Symposium on Computer and Information Sciences, pp. 71-87, Academic Press Inc., New York, N.Y.

Jackson, D. M. (1969b). Comparison of classifications. In: Cole (Ed.), Numerical Taxonomy, pp. 91-111, Academic Press Inc., Newyork, N.Y.

Jackson, D. M. (1970). The construction of retrieval experiments and pseudo-classification based on external relevance. Information Storage and Retrieval, 6, 187-219.

Jackson, D. M. (1971). Classification, relevance, and information retrieval. In: Alt and Rubinoff (Eds.); Yovits (Guest Ed.), Advances in Computers, Vol. 11, pp. 60-125, Academic Press Inc., Newyork, N. Y.

Jackson, D. M. and White, L. J. (1972). Stability problems in non-statistical classification theory. Computer J., 15, 214-221.

Jardine, N. Sibson, R. (1968a). The construction of hierarchic and non-hierarchic classifications. Computer J., 11, 177-184.

Jardine, N. and Sibson, R. (1968b). A model for taxonomy. Mathematical Biosciences, 2, 465-482.

Jardine, N. (1970). Algorithms, methods and models in simplification of complex data. Computer J., 13, 116-117.

Jardine, N. and Sibson, R. (1971). Mathematical Taxonomy. John Wiley & Sons, Inc., New York, N.Y.

Johnson, S. C. (1967). Hierarchical clustering schemes. Psychometrika, 12, 241-254.

Kendall, M. G. (1945). The Advanced Theory of Statistics. Vol. 1, C. Griffin and Co. Ltd., London.

Koontz, W. L. G., Narendra, P. M. and Fukunaga, K. (1976). A graph theoretic approach to non-parametric cluster analysis. IEEE Transactions on Computers, 25, 936-944.

Lance, G. N. and Williams, W. T. (1967a). A general theory of classificatory sorting strategies. I.Hierarchical systems. Computer J., 9, 373-382.

Lance, G. N. and Williams, W. T. (1967b). A general theory of classificatory sorting strategies. II. Clustering systems. Computer J., 10, 271-277.

Lesk, M. E. (1969). Word-word associations in document retrieval systems. American Documentation , 20, 27-38.

Lewis, P. A. W., Baxendale, P. B. and Bennett, J. L. (1967). Statistical discrimination of the synonymy / antonymy relationship between words. JACM., 14, 20-44.

Ling, R. F. (1972). On the theory and construction of k-clusters. Computer J., 15, 326-332.

Matula, D. W. (1977). Graph theoretic techniques for cluster analysis algorithms. In: Van Ryzin (Ed.), Advanced Seminar on Classification and Clustering, pp. 95-129, Academic Press Inc., Newyork, N. Y.

Minker, J., Wilson, G. A. and Zimmerman, B. H. (1972). An evaluation of query expansion by addition of clustered terms for a document retrieval system. Information Storage and Retrieval, 8, 329-348.

Minker, J. (1977). Information storage and retrieval- A survey and functional description. ACM-SIGIR Forum, Vol. XII (2), 1-108.

Morrison, D. G. (1974). Discriminant analysis. In:Ferber, Handbook of Marketing Research, pp. 2-443 to 2-457, Newyork, N. Y.

Mulligan, G. D. and Corneil, D. G. (1972). Corrections to Bierstone's algorithm for generating cliques. JACM., 19, 244-247.

Needham, R. M. and Sparck-Jones, K. (1964). Keywords and clumps. J. Documentation, 20, 5-15.

Needham, R. M. (1967). Automatic classification in linguistics. Statistician, 17, 45-54.

Nie, N. H., Hull, C. H., Jenkins, J. G., Steinbrenner, K. and Bent, D. H. (1975). SPSS: Statistical Package for Social Sciences , 2nd Edition, McGraw Hill, Newyork, N. Y.

Raghavan, V. V. and Yu, C. T. (1978). Experiments on the determination of the relationships between

terms (Abstract). In: Dattola (Ed.), Proceedings of the Internatinal Conference on Information Storage and Retrieval. ACM-SIGIR Forum, Vol. XIII (1), 150. The paper has been accepted for publication in ACM Transactions on Database Systems.

Rocchio, J. J. Jr. and Salton, G. (1965). Information search optimization and iterative techniques. Proceedings AFIPS. FJCC, Vol. 27, Pt. 1, Spartan Books, Newyork, 293-305.

Rocchio, J. J. Jr. (1966). Document retrieval systems- optimization and evaluation, doctoral thesis. In: Information Storage Retrieval, Scientific Report No. ISR-10, Harvard University, Cambridge, Mass.

Salton, G. (1966). Information dissemination and automatic information systems. Proceedings IEEE, 54, 12, 1663-1678.

Salton, G. and Lesk, M. E. (1968). Computer evaluation of indexing and text processing. JACM., 15, 8-36.

Salton, G. (1971). The Smart Retrieval System - Experiments in Automatic Document Processing. Prentice Hall, Englewood Cliffs, N. J.

Salton, G. (1972a). Experiments in automatic thesauras construction for information retrieval. In: Information Processing 71, North Holland Publishing Co., Amsterdam, 115-123.

Salton, G. (1972b). Comment on "An evaluation of query expansion by the addition of clustered terms for a document retrieval system". Information Storage and Retrieval, 8, 349.

Salton, G. (1975). Dynamic Information and Library Processing. Prentice-Hall, Englewood Cliffs, N. J.

Salton, G., Yang, C. S. and Yu, C. T. (1975). A theory of term importance in automatic text analysis. J. American Society for Information Science , 26, 33-44.

Senda, R. E. (1978). An Experimental Study of Term- Term Array Processing. M. Sc. Project, Department of Computing Science, The University of Alberta.

Sibson, R. (1970). A model for taxonomy-II.

Mathematical Biosciences, 6, 405-430.

Sneath, P. H. A. and Sokal, R. R. (1973). Numerical Taxonomy. Freeman, San Francisco, Ca.

Sokal, R. R. and Rohlf, F. J. (1962). The comparison of dendrograms by objective methods. Taxon, 11, 33-40.

Sparck-Jones, K. and Jackson, D. M. (1967). Current approach to classification and clump-finding at CLRU . Computer J. 10, 29-37.

Sparck-Jones, K. (1971). Automatic Keyword Classification for Information Retrieval. Archon Books, Connecticut.

Sparck-Jones, K. (1973). Collection properties influencing automatic term classification performance. Information Storage and Retrieval, 9, 499-513.

Stevens, M. E., Guiliano, V. E., and Heilprin, D. (Eds., 1965). Statistical Association Methods for Mechanised Documentation, Symposium Proceedings (1964), Miscellaneous Publication 269, National Bureau of Standards, Washington, D. C.

Stiles, H.E. (1961). The association factor in information retrieval. JACM., 8, 271-279.

Van Rijsbergen, C. J. (1971). An algorithm for information structuring retrieval. Computer J., 14, 407-412.

Vaswani, P. K. T. (1968). A technique for cluster emphasis and its application to automatic indexing. In: Information Processing 68, North Holland Publishing Co., Amsterdam, 1300-1303.

Vaswani, P. K. T. and Cameron, J. B. (1970). The NPL Experiment in Statistical Word Associations and Their Use in Document Indexing and Retrieval. Report Com. Sci. 42, National Physical Laboratory, Teddington, England.

Watanabe, S. (1972). A unified view of clustering algorithms. In: Information Processing 71, North Holland Publishing Co., Amsterdam, 149-154.

Williams, J. H. Jr. (1965). Results of classifying documents with multiple discriminant functions. In: Stevens, Guiliano, and Heilprin(Eds.), Statistical Association Methods for Mechanised

Documentation , Symposium Proceedings (1964), pp. 217-224, Miscellaneous Publication 269, National Bureau of Standards, Washington, D. C.

Wishart, D. (1969). A generalization of nearest neighbor which reduces chaining effects. In: Cole (Ed.), Numerical Taxonomy, pp. 282-311, Academic Press Inc., Newyork, N.Y.

Yu, C. T. (1974a). A methodology for the construction of term classes. Information Storage and Retrieval, 10, 243-251.

Yu, C. T. (1974b). A clustering algorithm based on user queries. J. of the American Society for Information Science, 25, 218-226.

Yu, C. T. (1975). A formal construction of term classes. JACM., 22, 17-37.

Yu, C. T., Luk, W. S. and Cheung, T. Y. (1976) A statistical model for relevance feedback in information retrieval. JACM., 23, 273-286.

Yu, C. T. (1976). The stability of two common matching functions in classification with respect to a proposed measure. J. of the American Society for Information Science, 27, 248-255.

Yu, C. T. and Raghavan, V. V. (1977). A single pass method for determining the semantic relationships between terms. J. American Society for Information Science , 28, 345-354.

Zahn, C. T. (1971). Graph theoretical methods for detecting and describing gestalt clusters. IEEE Transactions on Computers, C-20, 68-86.

APPENDIX - A

EXPERIMENTAL DETAILS FOR THE PROPOSED METHOD

A.1 Determination of the POS and NEG Counts

For each DR pair not satisfying assessment and for each DR pair satisfying assessment with the cosine function, but not with the new similarity function (3.1.2) based on the most current positive and negative counts, perform the following steps (Note: All the DR pairs of the former category are processed before any of the latter).

STEP 1. Identify the sets (D-R) and (R-D).

STEP 2. For every potential term pair $(t_i, t_j) \in$ (D-R) & (R-D), if D is relevant to R then increment $POS(t_i, t_j)$ by 1 else increment $NEG(t_i, t_j)$ by 1.

Futher details on the implementation of these steps are presented in (Senda, 1978).

A.2 The Strategy for Selecting the Evaluation Set of Queries

STEP 1. Assign all the queries to the base set.

STEP 2. Consider each query in turn. If every term in this query is present in at least one query (other than the one being considered)

in the base set, then assign the query to the evaluation set, and remove it from the base set.

Note that the result of this procedure may depend on the order in which the queries are processed.

A.3 The Incorporation of the Relationships Between Terms into the Retrieval Process

Consider the new similarity function (3.2.1). The details of the functions g_1 and g_2 are provided here.

Given D and R, let (t_i,t_j) be a potential term pair in (D-R) & (R-D). The change in the new similarity value due to this pair of terms is

$$= \begin{cases} 0 \text{ if } \mathcal{C} = 0 \\ (a_2 \,/\, |(D-R)|*|(R-D)|) * \mathcal{C}(t_i,t_j)*W(t_i,t_j) \text{ if } \mathcal{C} > 0 \\ (a_3 \,/\, |(D-R)|*|(R-D)|) * \mathcal{C}(t_i,t_j)*W(t_i,t_j) \text{ if } \mathcal{C} < 0 \end{cases}$$

where $W(t_i,t_j)$ is a weight of importance, between 0 and 1, associated with the pair of terms t_i and t_j and the divisor $|(D-R)| * |(R-D)|$ is a normalizing factor. The above form ensures that the overall effect of term relationship components on the modified similarity function is between $-a_3$ and $+a_2$. A number of alternative forms of W have been used, which we describe in the following appendices.

A.4 The Weighting Functions W_1 and W_2

The primary objective in the usage of weighting functions in our experiments is to develop an understanding of the relative significance of the relationships between various term pairs. In other words, the weight functions provide a mechanism by which different criteria for specifying the usefulness of the relationship between a pair of terms to the retrieval process can be tested.

The details of the weighting schemes are best described in reference to Figures 6 and 7. The figures represent a graph in which the axes are the values of positive count and k * negative count, where k is the balancing factor described in section 3.1.2. The term pairs on the line p = n have a ϵ value of zero and their relationships are, clearly, not significant. The pairs of terms in the shaded region (bounded by n = p and p = n * m + c) are assigned a weight of importance of at most w. But the pairs in the unshaded region which are farther away from the line p = n are considered more significant and are, therefore, assigned a weight of at least w.

Let p' denote the positive count and n' the negative count times k for some potential term pair (t_i,t_j). Further let p' > n'. Then, in reference to Figure 7, suppose (p',n') is between the lines p = n and p = n * m + c <p = n * m + c and n=∅>. Then, the larger the value of a_1/a $<d_1/d>$ the farther away the point is from the former line and the

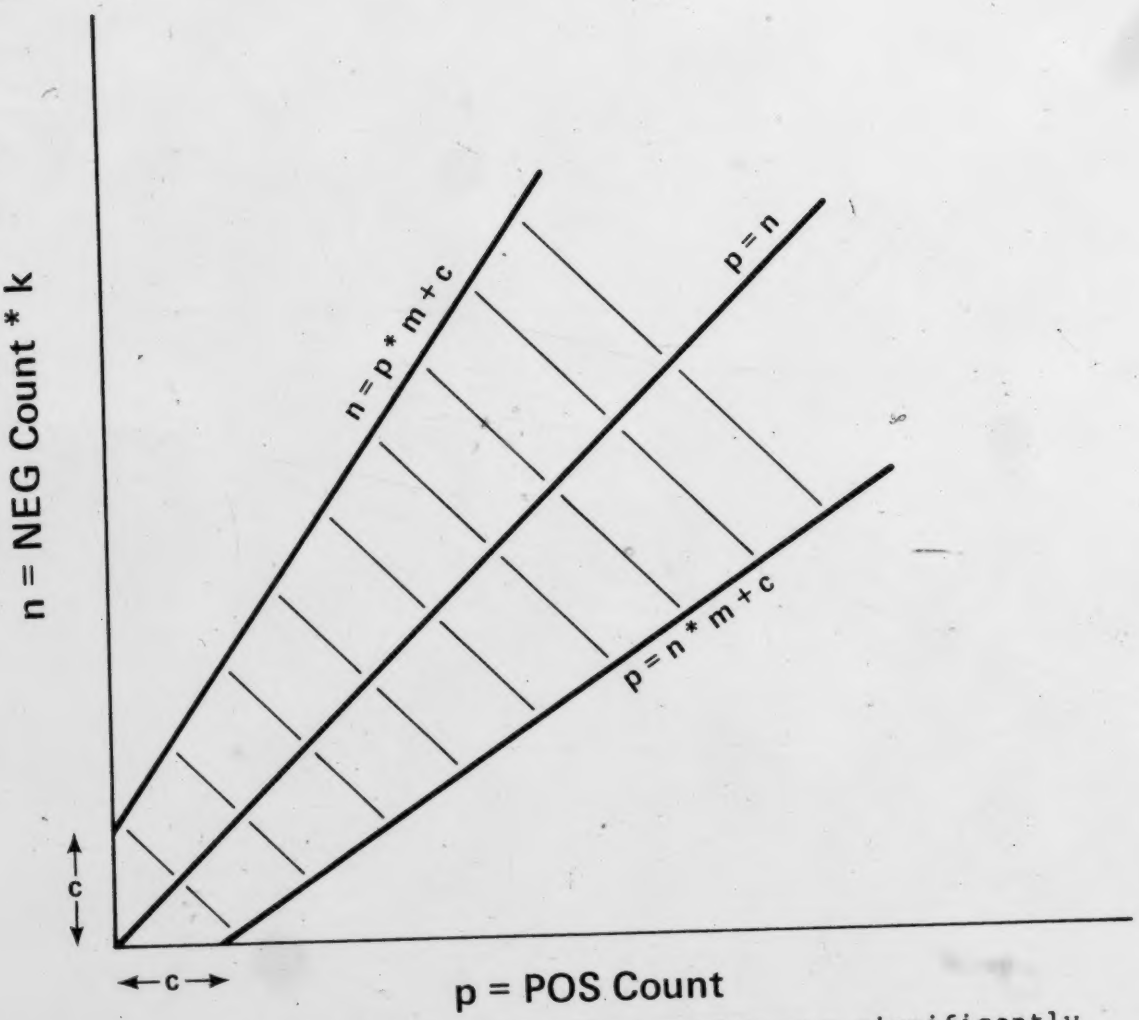


FIGURE 6. Representation of term pairs that are significantly positively or negatively related (unshaded region). m is a chosen gradient and c is a chosen constant.

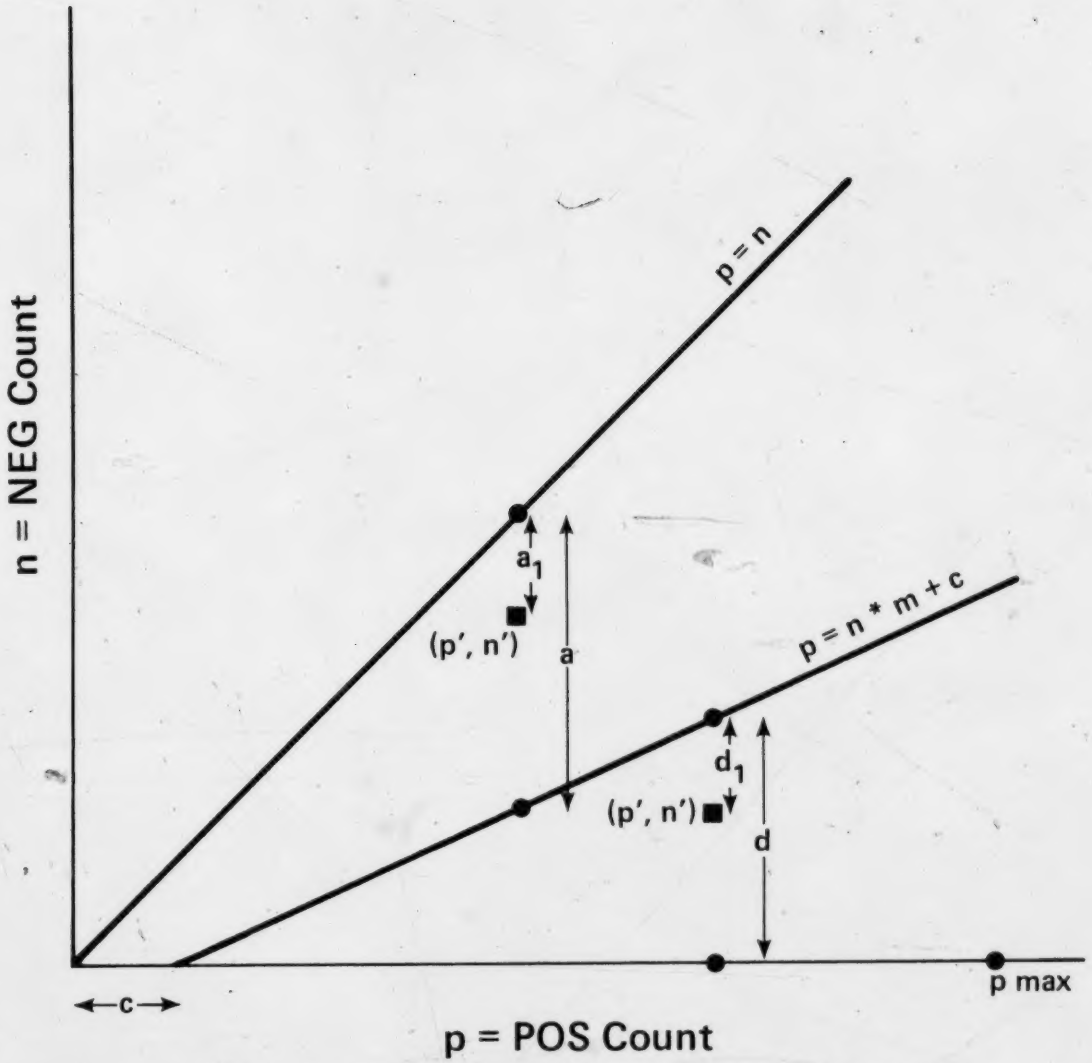


FIGURE 7. Illustration of the variables in the weighting functions. Point (p',n') corresponds to the positive and negative counts of some term pair. p_{max} is the largest positive count among all potential term pairs. m is a chosen gradient and c is a chosen constant.

closer it is to the latter line and, consequently, the more significant is the relationship between the corresponding term pair. Another factor that has some bearing on the significance is the ratio p'/p_{max}, where p_{max} is the largest positive count obtained over all potential term pairs. If two term pairs have the same value for a_1/a $\langle d_1/d \rangle$, but one term pair has a larger positive count, then more statistical evidence has been gathered for the larger case. Hence, the higher the ratio p'/p_{max} the greater is the significance attached. The function W_1 chosen meets the specifications described above.

$$W_1\ (t_i,t_j) = \begin{cases} w + (1-w)*(d_1/d)*(p'/p_{max}) \text{ if } (p',n') \text{ is in the unshaded region.} \\ w * (a_1/a)*(p'/p_{max}), \text{ otherwise.} \end{cases}$$

The weighting function W_2 is a particular case of W_1 where the factor p'/p_{max} is dropped for both the shaded and the unshaded regions. W_2 can be considered to place more credibility on the relationships obtained by our method than W_1 in the sense that the weights of importance are larger as a result of the factor dropped.

In the discussion above it is assumed that $p' > n'$. If, however, $p' < n'$ then the treatment is symmetric. For the experiments reported here, the value of the parameters w and

c are chosen to be Ø.25 and 1 respectively. The gradient m is taken to be 2 for the ADINUL collection and 3 for the CRN4NUL collection.

A.5 The Weighting Function W_3

The objective in devising the function W_3 is to appropriately incorporate the information concerning document frequencies of terms into the process of weight assignment. As terms with high document frequency are not used in the process of obtaining term relationships, the weighting function is specified as a function of the terms in the query that are acceptable for processing. Terms are considered to have high document frequency if they occur in more than (n/1Ø) documents, where n is the number of documents in the collection (Salton et al., 1975). Let the other terms be referred as low document frequency terms. Then, the function W is specified in the following way.

$$W_3\ (t_i, t_j) = \begin{cases} w + (1-w) * q * (d_1/d)*(p'/p_{max}) \text{ if } (p',n') \\ \qquad\qquad \text{is in the unshaded region.} \\ w * q * (a_1/a)*(p'/p_{max}), \text{ otherwise.} \end{cases}$$

where

$$q = \begin{cases} 1 \text{ if } (n/10) \geq x > 0.67 * (n/10) \\ 2 \text{ if } 0.67 * (n/10) \geq x > 0.33 * (n/10) \\ 3 \text{ if } 0.33 * (n/10) \geq x > 0 \end{cases}$$

and x is the average document frequency of the terms that are in the query and have low document frequency. Thus, the smaller the value of x in the range 0 to (n/10), the larger is the factor q. This feature favors the relationships between terms of low document frequency.

A.6 Method of Comparing Different Retrieval Strategies and the Computation of Statistical Significance

Given two methods, say A and B, the precision values at the various recall levels are computed for each query for the two methods. These measures are dependent on the size of the retrieved document set. Four additional measures, referred to as normalized recall, normalized precision, rank recall and log precision which are not dependent on the number of documents retrieved are also determined. Thus, the comparison of the effectiveness of the two methods is based on 11 of precision values (those at recalls 0,0.1, ...1), and the four global measures mentioned.

For each of the above measures the SMART system computes the probability of the average difference in the

measure between the methods being as large as that observed if in fact the two methods are equally good. Since it is difficult to judge a system's performance by using 15 different probability values, an aggregate probability value is computed. Assuming that the various individual tests are independent, the significance of this aggregate probability constitutes a single number index of the relative performances of the methods. If this aggregate probability does not exceed 0.001, then it may be concluded that most of the individual probabilities are smaller than 0.05 (Salton and Lesk, 1968). In other words, at this significance level, for the majority of the evaluation measures on which the two methods are compared there is less than 1 out of 20 chances of the average difference being as large. Thus, the null hypothesis that the two methods are equally good may be rejected.

The SMART system provides a number of alternatives for computing these probabilities. The tests available are the t-test, the sign-test and the Wilcoxon signed rank test. The t-test and the Wilcoxon tests take information concerning the magnitude of the differences in test statistics into account. Thus, they require assumptions about the distribution of the underlying data. The sign test, however, considers only the sign of the differences, and the probability of observing a positive difference is taken to be the same as that of observing a negative difference (1/2). Since this assumption is less demanding, the

discussions in section 3.6 are based on the probabilities determined using the sign test.

APPENDIX - B

TECHNICAL RESULTS RELATING TO STABILITY ANALYSIS

Consider the problem:

Given L_j, $1 \leq j \leq n$, such that $L_1 > L_2 > \cdots L_n$,

$$\text{maximize } \sum_{j=1}^{n} (x_j^2 - x_j) \tag{1}$$

subject to the constraints $\sum_{j=1}^{n} x_j = c$, for some constant $c \leq \sum_{j=1}^{n} L_j$ and $\emptyset \leq x_j \leq L_j$ for each j.

Lemma B.1: Let $c = \sum_{j=1}^{i-1} L_j + b$, for some i, $1 \leq i \leq n$, and some b, $1 \leq b < L_i$. Then the following is a maximum solution to (1).

$$x_j = L_j \text{ for } 1 \leq j \leq i-1,\ x_i = b, \text{ and}$$
$$x_j = \emptyset \text{ for } i+1 \leq j \leq n. \tag{2}$$

Remark: It is clear that both $(x_1, \cdots x_s, \cdots x_t, \cdots x_n)$ with $x_s = \emptyset$ and $x_t \neq \emptyset$ and $(x_1, \cdots x_t, \cdots x_s, \cdots x_n)$ have the same

value for $\sum_{j=1}^{n} (x_j^2 - x_j)$. Thus, we shall assume that all non-zero values must occur in the beginning.

Proof: Suppose (2) is not a maximum. Then, let

$$y_1, y_2, \cdots, y_n \tag{3}$$

be a maximum solution different from (2). Let s be the smallest j such that $y_j < L_j$. Let $t > s$ be the

smallest j such that $y_j > 0$.

Case(i). $y_s \geq y_t$

Let $y_s' = y_s + 1$ and $y_t' = y_t - 1$. Then, by simple computation it can be shown that

$(y_1, \ldots y_{s-1}, y_s', y_{s+1}, \ldots y_{t-1}, y_t', y_{t+1}, \ldots y_n)$

is a better solution to (1) than (3), contradicting the maximality of (3).

Case(ii). $y_s < y_t$

$y_s \neq 0$ by remark.

Let a new solution be formed with $y_s' = y_t + 1$ and $y_t' = y_s - 1$. None of the constraints are violated as $L_s > L_t$ and $y_t \leq L_t$. Again, it is a simple matter to show that the solution with y_s' and y_t' replacing y_s and y_t is better than (3). #

Corollary B.2: Suppose problem (1) is modified so that $\sum_{j=1}^{n} x_j \leq c = \sum_{j=1}^{i-1} L_j + b$. Then, solving this modified problem is equivalent to solving (1) since $\sum_{j=1}^{n} (x_j^2 - x_j)$ is a non-decreasing function. Furthermore, the maximum value is

$$\{ \sum_{j=1}^{i-1} L_j^2 + b^2 - \sum_{j=1}^{i-1} L_j - b \}.$$

Lemma B.3: Given a complete graph G in n vertices, suppose i edges are deleted from G. If f(n, i) is the minimum number of edge additions required to make the resulting graph connected, then

$$f(n,i) \leq 1 + (4i/(2n-1)).$$

Proof: It is clear that f(n,i) will be the maximum if the deletions are made in such a way that the perturbed graph has the largest number of components.

From lemma 5.3.4 we can deduce that a graph in C(n,2) - i edges and n vertices that has the maximum number of components has (n - k) isolated verties and a partially complete subgraph in the remaining k vertices, if $C(k,2) + 1 \leq C(n,2) - i \leq C(k+1,2)$. Thus, $f(n,i) \leq n - k$. Expressing k as a function of n and i, we have

$$f(n,i) \leq n - \{\lceil \{-1 + \mathrm{SQRT}[1 + 4(n(n-1) - 2i)]\}/2 \rceil\}$$

$$\leq n + 1/2 - \mathrm{SQRT}(1 + 4n^2 - 4n - 8i)/2$$

$$= n + 1/2 - \mathrm{SQRT}((2n-1)^2 - 8i)/2$$

$$= n + 1/2 - \mathrm{SQRT}\ (2n - 1 - 2\mathrm{SQRT}2i)(2n - 1 + 2\mathrm{SQRT}2i)/2$$

$$\leq n + 1/2 - (4n^2 - 4n - 8i + 1)/(4n - 2)$$

since $\mathrm{SQRT}(a.b) \geq 2ab/(a+b)$ for $a, b > \emptyset$

$$= 1 + 4i/(2n-1).$$

$$\simeq 1 + i(2/n)$$

#